



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΣΧΟΛΗ ΟΙΚΟΝΟΜΙΑΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Πτυχιακή Εργασία

Σχεδιασμός και ανάπτυξη τηλεματικής εφαρμογής
ηλεκτρονικής ζυγαριάς (υλικό και λογισμικό) με
εφαρμογή στην μελισσοκομία.

Κωνσταντίνος Σωτήριος Σωφρονάς
A.M.: 2022201600167

Επιβλέπων:
Νικόλαος Τσελίκας
Αναπληρωτής Καθηγητής

Τρίπολη, Δεκέμβριος 2022

Περιεχόμενα

Πνευματικά δικαιώματα	5
Ευχαριστίες	6
Κατάλογος σχημάτων	9
Περίληψη	10
Abstract	11
1 Εισαγωγή & Σκοπός Πτυχιακής Εργασίας	12
1.1 Εισαγωγή	12
1.2 Σκοπός	13
2 Τεχνολογίες Λογισμικού - Software	14
2.1 Arduino	14
2.2 Website	15
2.2.1 Xampp	15
2.2.1.1 Apache Server	17
2.2.1.2 MariaDB - MySQL Server	17
2.2.1.3 FileZilla	18
2.2.1.4 Mercury	19
2.2.1.5 Tomcat	19
2.2.2 HTML 5	19
2.2.3 CSS	20
2.2.3.1 Bootstrap	20
2.2.4 PHP	21
2.2.5 JavaScript	21
2.2.5.1 Ajax	22
2.3 Android Studio	22
3 Τεχνολογίες Υλικού - Hardware	23
3.1 Arduino Uno Rev 3	23
3.2 Arduino GSM Shield 2 (antenna connector)	25
3.3 Προέκταση κεραίας GSM	26
3.4 Αισθητήρας Θερμοκρασίας DS18B20	27
3.5 Αισθητήρας Υγρασίας DHT11	28
3.6 Αντίσταση 4.7 & 10 Kohm	29

3.7	Μετατροπέας 24-bit A/D Conversion with HX711	29
3.8	Αισθητήρας βάρους αντοχής 150 kg	30
3.9	Jumper Cables	31
3.10	Breadboard 400 οπών (Half size)	31
3.11	Μπαταρία 12 Volt 7.2Ah	32
3.12	Αντάπτορας για τροφοδότηση του Arduino	33
3.13	Φωτοβολταϊκός Συλλέκτης 15W - 12V	33
3.14	Ρυθμιστής Φόρτισης 20Ah 12Volt - 24Volt	34
3.15	Στεγανά Κουτιά	34
3.16	Κύκλωμα Timer Relay	35
3.17	Διακόπτες on/off	36
3.18	Χειροποίητο κουτί κατασκευής	36
3.19	Κοστολόγιο υλικών	38
4	Σχεδιασμός του Συστήματος	39
4.1	Δεδομένα	39
4.2	Ζητούμενα	39
4.2.1	Γενική περιγραφή λειτουργιών του συστήματος	39
4.2.2	Κατηγορίες χρηστών	40
4.2.3	Περιβάλλον λειτουργίας	40
4.2.4	Περιορισμοί στην σχεδίαση και την υλοποίηση	40
4.3	Λειτουργίες Συστήματος	41
4.3.1	Πλοήγηση επισκέπτη	41
4.3.2	Αλλαγή στοιχείων χρήστη - μελισσοκόμου στο σύστημα	42
4.3.3	Εγγραφή νέου χρήστη - μελισσοκόμου	43
4.3.4	Εισαγωγή χρήστη - μελισσοκόμου στο σύστημα	44
4.3.5	Αλλαγή στοιχείων χρήστη - μελισσοκόμου στο σύστημα	45
4.3.6	Προβολή δεδομένων από την ηλεκτρονική ζυγαριά	46
4.3.7	Λειτουργίες διαχειριστή	47
4.3.8	Καταγραφή Περιπτώσεων Χρήσης	48
4.3.8.1	Περίπτωση Χρήσης #1	48
4.3.8.2	Περίπτωση Χρήσης #2	50
4.3.8.3	Περίπτωση Χρήσης #3	52
4.3.8.4	Περίπτωση Χρήσης #4	55
5	Υλοποίηση του Συστήματος	58
5.1	Arduino	58
5.1.1	Απαραίτητες Βιβλιοθήκες	58

5.1.2	Τρόπος Εγκατάστασης Βιβλιοθηκών	58
5.1.3	Βιβλιοθήκες	61
5.1.4	Συνδεσμολογία Αισθητήρων με το Arduino	62
5.1.4.1	Συνδεσμολογία DS18B20	62
5.1.4.2	Συνδεσμολογία DHT11	63
5.1.4.3	Συνδεσμολογία Αισθητήρα Βάρους - Μετατροπέα HX711	65
5.1.4.4	Συνδεσμολογία - GSM Shield	66
5.1.4.5	Σχεδιάγραμμα Συνδέσεων Αισθητήρων	67
5.1.4.6	Τελική Συνδεσμολογία Όλων των Υλικών	68
5.1.5	Κώδικας Δοκιμής	69
5.1.5.1	Δοκιμή DS18B20	69
5.1.5.2	Δοκιμή DHT11:	70
5.1.5.3	Δοκιμή HX711:	71
5.1.5.4	Δοκιμή GSM Shield	73
5.1.6	Τελική Μορφή Κώδικα	75
5.2	Website	84
5.2.1	Δημιουργία Βάσης Δεδομένων	84
5.2.2	Πρόγραμμα Παραγωγής τυχαίων δεδομένων	89
5.2.3	Πρόσβαση Στην Ιστοσελίδα	93
5.2.4	Δομή Καταλόγου	94
5.2.4.1	Αρχείο ρυθμίσεων σύνδεσης βάσης δεδομένων	95
5.2.4.2	Αρχείο κεντρικής σελίδας	96
5.2.4.3	Επεξήγηση δομής θέματος αρχικής σελίδας	98
5.2.4.4	Αρχείο Πληροφοριών	110
5.2.4.5	Αρχείο Επικοινωνίας	111
5.2.4.6	Αρχείο Εγγραφής Χρήστη	115
5.2.4.7	Αρχείο Σύνδεσης Χρήστη	135
5.2.4.8	Επεξεργασία στοιχείων Προφίλ Χρήστη	142
5.2.4.9	Περιβάλλον Προβολής δεδομένων	147
5.2.4.10	Αποσύνδεση Χρήστη Από το Σύστημα	156
5.2.4.11	Παρουσίαση Συστήματος από κινητή συσκευή	158
5.3	Android - Τηλεματική Εφαρμογή	160
5.3.1	Δομή Αρχείων	160
5.3.1.1	Αρχείο προβολής πριν το ξεκίνημα της εφαρμογής . . .	161
5.3.1.2	Αδυναμία Πρόσβασης στο Internet	166
5.3.1.3	Αρχείο φόρμα σύνδεσης	168
5.3.1.4	Αρχείο κεντρικής προβολής	173
5.3.1.5	Προβολή: Σήμερα	177

5.3.1.6	Προβολή: Εχθές	183
5.3.1.7	Προβολή: Εβδομάδα	189
5.3.1.8	Προβολή: Μήνας	197
5.3.1.9	Προβολή: Αναζήτηση Ημέρας	211
5.3.1.10	Μενού Επιλογών	222
5.3.1.11	Προβολή: Ρυθμίσεις	223
5.3.1.12	Προβολή: Πληροφορίες	224
5.3.1.13	Αρχείο Ενημέρωσης Δεδομένων Βάσης Δεδομένων . .	226
5.3.2	Δομή Αρχείων Επικοινωνίας Εφαρμογής με την Βάση Δεδομένων	231
6	Συμπεράσματα / Μελλοντικές Επεκτάσεις	237
6.1	Συμπεράσματα	237
6.2	Μελλοντικές Επεκτάσεις	237
Αναφορές		239

Πνευματικά δικαιώματα

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πελοποννήσου.

Copyright ©, Πανεπιστήμιο Πελοποννήσου, 2022

Eυχαριστίες

Θα ήθελα να ευχαριστώ θερμά τον διδάσκοντα και επιβλέποντα της πτυχιακής εργασίας, κ. Νικόλαο Τσελίκα για την ευκαιρία και τη πολύτιμη στήριξη και καθοδήγησή του, καθώς και όλα τα άτομα που συνέβαλαν στην υλοποίηση και την ολοκλήρωση αυτής της πτυχιακής εργασίας.

Ένα ακόμη ευχαριστώ σε όλους τους καθηγητές μου, που με τις γνώσεις τους συνέβαλαν στη διαμόρφωση των θεωρητικών, πρακτικών και τεχνολογικών γνώσεών μου.

Τέλος, θα ήθελα να ευχαριστήσω τους χοντινούς μου ανθρώπους (γονείς και φίλους), που όλα αυτά τα χρόνια είναι χοντά μου και με στηρίζουν σε κάθε μου βήμα.

Κατάλογος Σχημάτων

1	Περιβάλλον - Arduino IDE	14
2	Хамрр - Πίνακας ελέγχου	15
3	Κεντρική Σελίδα - Хамрр	16
4	Λογότυπο - Apache Http Server	17
5	Λογότυπο - MariaDB	17
6	Διαχειριστικό Περιβάλλον - SQL μέσω PhpMyAdmin	18
7	Λογότυπο - FileZilla	18
8	Λογότυπο - Mercury	19
9	Λογότυπο - Tomcat	19
10	Λογότυπο - HTML 5	19
11	Λογότυπο - CSS 3	20
12	Λογότυπο - Bootstrap	20
13	Λογότυπο - PHP	21
14	Λογότυπο - JavaScript	21
15	Λογότυπο - AJAX	22
16	Περιβάλλον - Android Studio IDE	22
17	Arduino Uno Rev 3	23
18	Διάταξη ακίδων του Arduino Uno Rev 3	24
19	GSM Shield 2	25
20	Διάταξη ακίδων της GSM Shield Antenna 2	26
21	Κεραία GSM	26
22	Αισθητήρας Θερμοκρασίας DS18B20	27
23	Αισθητήρας υγρασίας DHT11 Αριστερά (4 pins - απαιτείται η χρήση αντίστασης) Δεξιά (3 pins - έτοιμη πλακέτα)	28
24	Resistor 4.7 kΩ (στα αριστερά) και Resistor 10 kΩ (στα δεξιά)	29
25	HX 711	29
26	Weight Sensor 150 kg	30
27	Jumper Cables	31
28	Breadboard 400 θέσεων	32
29	Μπαταρία 12Volt 7.2Ah	32
30	Αντάπτορας Μπαταρίας	33
31	Φωτοβολταϊκός Συλλέκτης 15W - 12V	33
32	Ρυθμιστής Φόρτισης Μπαταρίας 20A	34
33	Κουτί Στεγανό	34
34	Timer Relay	35
35	Διακόπτης on/off	36

36	Χειροποιήτο κουτί αποθήκευσης ηλεκτρονικής ζυγαριάς	36
37	Ηλεκτρονική Ζυγαριά - Τοποθετημένη Κυψέλη	37
38	Βήμα 1o - Εγκατάσταση Βιβλιοθήκης - Arduino	59
39	Βήμα 2o - Αναζήτηση και Επιλογή Έκδοσης Βιβλιοθήκης - Arduino . .	59
40	Συνδεσμολογία: DS18B20 - Arduino	62
41	Συνδεσμολογία: DHT11 (4 pin) - Arduino	63
42	Συνδεσμολογία: DHT11 (3 pin) - Arduino	64
43	Συνδεσμολογία: Αισθητήρα Βάρους - Μετατροπέα HX711 - Arduino .	65
44	Συνδεσμολογία: GSM Shield - Arduino	66
45	Σχεδιάγραμμα Αισθητήρων (Τελική Μορφή)	67
46	Σχεδιάγραμμα Κατασκευής (Τελική Μορφή)	68
47	Δημιουργία Βάσης Δεδομένων	84
48	Όνομα και Τύπος Βάσης Δεδομένων	85
49	Δημιουργία Νέας Βάσης Δεδομένων	85
50	Επιλογή ονόματος και αριθμός στηλών πίνακα data	86
51	Επιλογή δομής πίνακα data	87
52	Επιλογή δομής πίνακα users	88
53	Μορφή αρχείου .csv	91
54	Επιβεβαίωση Εισαγωγής Δεδομένων από το αρχείο .csv	92
55	Κεντρική Σελίδα	93
56	Θέμα Πρώτο	97
57	Θέμα Πρώτο (Συνδεδεμένος Χρήστης)	97
58	Θέμα Δεύτερο	98
59	Θέμα Πρώτο - Πρώτη περιοχή (container)	98
60	(Εικόνα Αριστερά) - Θέμα Πρώτο - Πρώτη περιοχή (Συνδεδεμένος Χρήστης)	99
61	(Εικόνα Δεξιά) - Θέμα Πρώτο - Πρώτη περιοχή (Μενού Χρήστη Αρχικής Σελίδας)	99
62	Θέμα Πρώτο - Δεύτερη περιοχή (Μπάρα Πλοήγησης)	99
63	Θέμα Πρώτο - Τρίτη περιοχή (Περιεχόμενο σελίδας)	100
64	Θέμα Πρώτο - Τέταρτη περιοχή (Footer)	100
65	Σελίδα Πληροφοριών	110
66	Σελίδα Επικοινωνίας	111
67	thankyou.php - Σελίδα Ευχαριστιών	114
68	error.php - Σελίδα Σφάλματος	114
69	register.php - Σελίδα Εγγραφής	115
70	registered.php - Σελίδα Επιτυχούς Εγγραφής Χρήστη	134
71	Email - Στοιχείων Εγγραφής Χρήστη	134

72	login.php - Σελίδα Σύνδεσης	136
73	Αλλαγή σελίδας μετά τη σύνδεση του χρήστη	141
74	Προφίλ Χρήστη	142
75	Αρχική Σελίδα Παρουσίασης Δεδομένων	147
76	Μήνυμα Μη ύπαρξης Δεδομένων	147
77	Παρουσίαση Δεδομένων Προηγούμενης Ημέρας	148
78	Μήνυμα Μη ύπαρξης Δεδομένων Χθεσινής Ημέρας	148
79	Παρουσίαση Δεδομένων Τρέχουσας Εβδομάδας	149
80	Μήνυμα Μη ύπαρξης Δεδομένων Εβδομάδας	149
81	Μενού Επιλογής Μήνα	150
82	Δεδομένα Ιουλίου	151
83	Μήνας χωρίς δεδομένα (Οκτώβριος)	151
84	Αναζήτηση Συγκεκριμένης Ημέρας	154
85	Επιλογή Συγκεκριμένης Ημέρας	154
86	Αποτέλεσματα Συγκεκριμένης Ημέρας	155
87	Αποσύνδεση	156
88	Ιστοσελίδα μέσω Κινητής Συσκευής	158
89	Ιστοσελίδα μέσω Κινητής Συσκευής - Μέρος 2ο	159
90	Αρχική Προβολή Φόρτωσης Εφαρμογής	161
91	Προβολή Αδυναμίας σύνδεσης Internet	166
92	Προβολή φόρμας εισόδου χρήστη	168
93	Μενού Επιλογών Κάτω	176
94	Προβολή Δεδομένων - Σήμερα	177
95	Προβολή Δεδομένων - Εχθές	183
96	Προβολή Δεδομένων - Εχθές	189
97	Προβολή Δεδομένων - Μήνα	198
98	Προβολή Δεδομένων - Αναζήτηση Συγκεκριμένης Ημέρας	212
99	Μενού επιλογών - Επάνω δεξιά	222
100	Προβολή Ρυθμίσεις	223
101	Προβολή Πληροφοριών	224

Περίληψη

Στη παρούσα πτυχιακή εργασία υλοποιείται ένα σύστημα παρακολούθησης - ηλεκτρονική ζυγαριά - μέσω τηλεματικής εφαρμογής, με εφαρμογή στην μελισσοκομία. Ο εκάστοτε χρήστης - μελισσοκόμος - θα μπορεί να παρακολουθεί μέσα από την εφαρμογή, τα δεδομένα για τη πορεία ανάπτυξης του μελισσοκομικού λαού.

Η ηλεκτρονική ζυγαριά περιέχει έναν μικροελεγκτή arduino που ελέγχει μια πλακέτα επικοινωνίας gsm (gsm shield), αισθητήρες βάρους, θερμοκρασίας και υγρασίας. Λειτουργεί κάθε δώδεκα ώρες με χρονοδιαχόπτη για εξοικονόμηση ενέργειας. Η παροχή ρεύματος γίνεται μέσω μπαταρίας και η φόρτιση της πραγματοποιείται μέσω συστήματος που εκμεταλλεύεται την ηλιακή ενέργεια (ηλιακό πάνελ).

Η χρήση της ηλεκτρονικής ζυγαριάς γίνεται για τη συλλογή δεδομένων και αποστολή τους, με γραπτό μήνυμα, sms. Στη συνέχεια, τα δεδομένα διαβάζονται από την android εφαρμογή και στέλνονται σε βάση δεδομένων ώστε να γίνει η αποθήκευση τους.

Έτσι, ο μελισσοκόμος έχει πρόσβαση ανά πάσα χρονική στιγμή σε δεδομένα, όπως το βάρος, η υγρασία και η θερμοκρασία, της κυψέλης με την οποία έχει διασυνδεθεί η ηλεκτρονική ζυγαριά. Τα αποτελέσματα που συλλέγονται τον βοηθούν στον απομακρυσμένο έλεγχο και στην αποδοτικότερη διαχείριση των κυψελών του.

Λέξεις - Κλειδιά: Android, Εφαρμογή, Ηλεκτρονική ζυγαριά, Arduino, GSM, Κινητό δίκτυο, Java, PHP, Javascript, JSON, SQL, Android Studio, C, Ενσωματωμένα Συστήματα

Abstract

In this thesis a monitoring system is implemented - electronic scale - through telematics application with application in beekeeping. Through this application, the beekeeper-user, will be able to monitor critical data regarding his/her hive and its growth.

The electronic scale, contains an arduino microcontroller that controls a gsm shield, weight, temperature and humidity sensors, lights up every twelve hours using a timer relay to save energy. The power supply is achieved with a battery supported by a solar panel.

The electronic scale is used to collect useful data and send it via sms to the end-user. This data is also received by the android application, which is then responsible to save it in a data base as well to present it in a more user-friendly way to the end-user.

Thus, the beekeeper is able to see at any time both current and past data, such as the weight, the humidity and the temperature of the hive and based on this data he/she can take critical decisions or follow specific actions, resulting to a better management of beekeeping.

Keywords: Android, Application, Electronic scales, Arduino, GSM, Mobile Network, Java, PHP, Javascript, JSON, SQL, Android Studio, C, Embedded Systems

Κεφάλαιο 1ο

Εισαγωγή & Σκοπός Πτυχιακής Εργασίας

1.1 Εισαγωγή

Η μελισσοκομία είναι η τέχνη της εκτροφής μελισσών. Στη χώρα μας έχει αρκετά παλιά παράδοση που κρατά από την αρχαιά Ελλάδα. Υπάρχουν δύο ειδών άσκησης της μελισσοκομίας, η στατική και η νομαδική. Ο όρος στατική μελισσοκομία είναι η τοποθέτηση των κυψελών σε ένα μοναδικό μέρος, ενώ ο όρος νομαδική μελισσοκομία, η συχνή μετακίνηση των κυψελών από μέρος σε μέρος, με σκοπό την εξασφάλιση κατάλληλων συνθηκών για την διαβίωση τους.

Κατά κύριο λόγο στην Ελλάδα, επικρατεί η νομαδική μελισσοκομία. Ανάλογα την κάθε εποχή, υπάρχει και η αντίστοιχη ανθοφορία. Δυστυχώς όμως, σε μεγάλο μέρος της χώρας τα ποσοστά ανθοφορίας δεν επαρκούν για τη συντήρηση του μελισσοκομικού λαού και αυτό έχει οικονομικό αντίκτυπο για το μελισσοκόμο. Με σκοπό να δωθεί η δυνατότητα στο λαό, να εκμεταλλευτεί την αντίστοιχη ανθοφορία, της εκάστοτε περιοχής, για την ανάπτυξη του, ο μελισσοκόμος αναγκάζεται να μεταναστεύσει το λαό του σε διαφορετικά μέρη. Αυτή η ενέργεια περιέχει μεγάλο ρίσκο και ίσως μη επιθυμητά αποτελέσματα. Για παράδειγμα, οι καιρικές συνθήκες της περιοχής μπορεί να επιδράσουν κατασταλτικά στην εξέλιξη των μελισσιών.

Έτσι, δημιουργήθηκε η ανάγκη απομακρυσμένης παραχολούμενης, δηλαδή, η λήψη δεδομένων για τα καιρικά φαινόμενα που επικρατούν στη περιοχή και για το βάρος του μελισσοκομικού λαού. Δια μέσου του βάρους, μπορεί να διαπιστωθεί με γνώμονα τις μεταβολές του, η πορεία του, δίνοντας έτσι μια εμπειριστατωμένη εικόνα για τις ανάγκες του μελισσιού.

Με την ανάπτυξη της ηλεκτρονικής ζυγαριάς παρέχεται η δυνατότητα συλλογής δεδομένων στο μελισσοκόμο για τα μελίσσια του, σε άμεσο χρόνο, επιτρέποντας του τη μείωση των περιττών μετακινήσεων. Αυτό συνεπάγεται και στη μείωση των εξόδων, καθώς και την άμεση επέμβαση του όταν αυτό χρίνεται αναγκαίο.

1.2 Σκοπός

Σκοπός της πτυχιακής, είναι η δημιουργία μιας ηλεκτρονικής ζυγαριάς, τόσο σε επίπεδο κατασκευαστικό (hardware), όσο και σε επίπεδο λογισμικού (software). Η ζυγαριά αυτή θα στέλνει στο χρήστη, με γραπτό μήνυμα (SMS), σημαντικές μετρήσεις που έχει λάβει σχετικά με τη θερμοκρασία, την υγρασία του περιβάλλοντος αλλά και το βάρος του μελισσού.

Με αυτές τις μετρήσεις, δίνεται η δυνατότητα στο χρήστη να γνωρίζει άμεσα τις συνθήκες που επικρατούν στο μικροκλίμα της περιοχής και έτσι, να λαμβάνει τις απαραίτητες αποφάσεις για μελλοντικές ενέργειες που πρέπει να προβεί στη μελισσοκομική του μονάδα.

Σκοπός ακόμη, είναι η ανάπτυξη μιας τηλεματικής εφαρμογής, η οποία θα διαβάζει κατάλληλα το γραπτό μήνυμα (SMS) και θα αποθηκεύει τα δεδομένα σε μια βάση δεδομένων. Με αυτή τη λειτουργία υπάρχει άμεσα η δυνατότητα προβολής των δεδομένων σε πλατφόρμα διαδικτυακής πρόσβασης, αλλά και η δυνατότητα προβολής μέσω της ήδη υπάρχουσας τηλεματικής εφαρμογής.

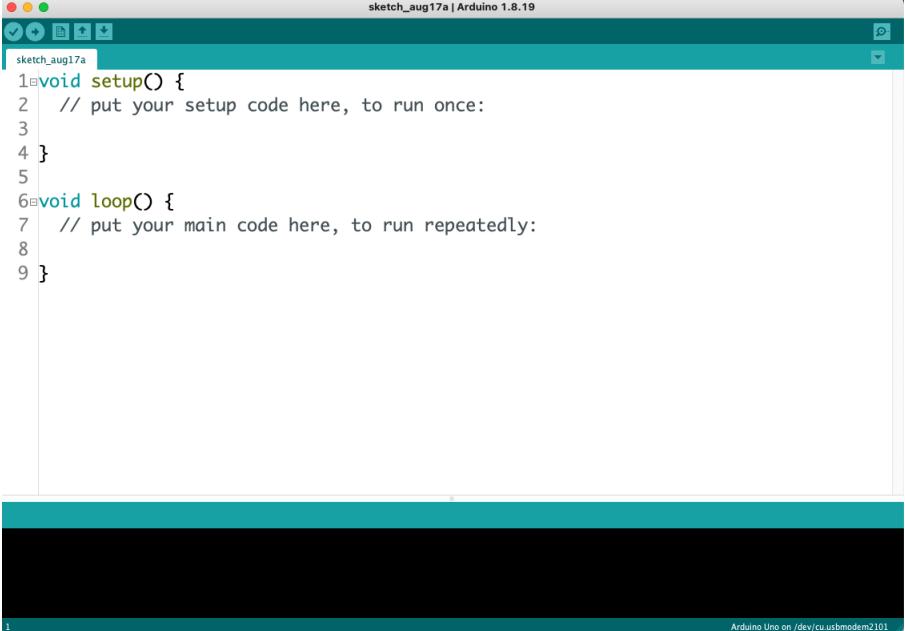
Κεφάλαιο 2ο

Τεχνολογίες Λογισμικού - Software

Σε αυτό το κεφάλαιο, θα γίνει ανάλυση των τεχνολογιών που χρησιμοποιήθηκαν στο τομέα του λογισμικού (software).

2.1 Arduino

Για την ανάπτυξη του κώδικα της ηλεκτρονικής ζυγαριάς, χρησιμοποιήθηκε το πρόγραμμα Arduino IDE, που προσφέρει, το Arduino. Η γλώσσα προγραμματισμού που απαιτείται για τη συγγραφή προγραμμάτων, είναι μια απλή έκδοση της C, ονομάζοντας την, Wiring C.



```
sketch_aug17a | Arduino 1.8.19
sketch_aug17a
1void setup() {
2 // put your setup code here, to run once:
3
4 }
5
6void loop() {
7 // put your main code here, to run repeatedly:
8
9 }
```

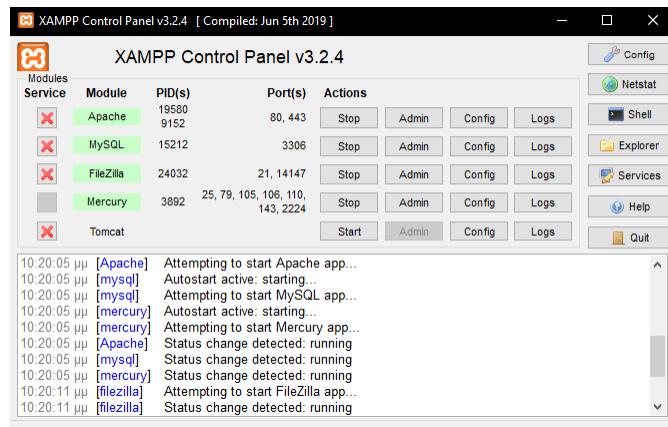
Arduino Uno on /dev/cu.usbmodem2101

Σχήμα 1: Περιβάλλον - Arduino IDE

2.2 Website

Για την ανάπτυξη του ιστότοπου, ήταν αναγκαίο να χρησιμοποιηθούν προγράμματα και γλώσσες προγραμματισμού που παρουσιάζονται παραχάτω:

2.2.1 Xampp



Σχήμα 2: Xampp - Πίνακας ελέγχου

Το Xampp, είναι ένα σύνολο προγραμμάτων, ανοικτού κώδικα και χρησιμοποιείται ως πλατφόρμα ανάπτυξης ιστοσελίδων και βάσεων δεδομένων. Ο σκοπός ανάπτυξης της εφαρμογής, ήταν να βοηθήσει τους προγραμματιστές να μην αντιμετωπίζουν προβλήματα κατά την εγκατάσταση των προγραμμάτων ξεχωριστά, καθώς είναι αρκετά δύσκολο και χρονοβόρο.

Το πακέτο προγραμμάτων που περιέχει το Xampp, είναι ο Apache Server, Maria DB database, interpreters για προγράμματα που είναι γραμμένα σε γλώσσες προγραμματισμού, όπως είναι η PHP και η Perl.

Το πρόγραμμα προσφέρεται σε διανομές των Windows, Linux, OS X. Το επίσημο site του, βρίσκεται στη διεύθυνση: <https://www.apachefriends.org/index.html>. Η πρόσβαση στο κεντρικό πίνακα ελέγχου γίνεται μέσω της ιστοσελίδας: <http://localhost/dashboard/> από οποιοδήποτε πλοηγητή.



Σχήμα 3: Κεντρική Σελίδα - Xampp

Για τη δημιουργία ελέγχου και δοκιμής ενός site αρχεί να τοποθετηθούν τα αρχεία στην εγκατάσταση του φακέλου του Xampp στο φάκελο htdocs.

Windows

C:\xampp\htdocs

MacOS

/Applications/XAMPP/xamppfiles/htdocs

2.2.1.1 Apache Server



Σχήμα 4: Λογότυπο - Apache Http Server

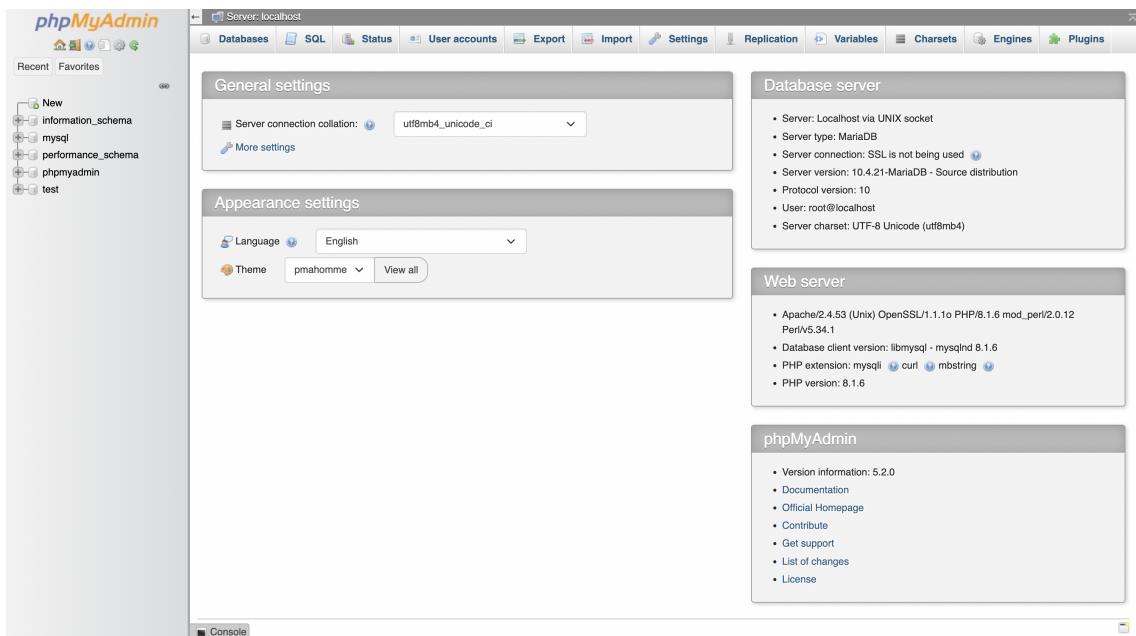
Πρόκειται για ένα σημαντικό εργαλείο που απαιτείται για την άναπτυξη εφαρμογής, που σχετίζεται με το διαδίκτυο. Επιτρέπει σε έναν υπολογιστή να φιλοξενήσει ένα ή περισσότερα site και να είναι προσβάσιμο είτε από το τοπικό δίκτυο (local network) είτε από το Internet.

2.2.1.2 MariaDB - MySQL Server



Σχήμα 5: Λογότυπο - MariaDB

Για την υποστήριξη των βάσεων δεδομένων χρησιμοποιείται η MariaDB. Ο MySQL Server αποτελεί ένα σύστημα διαχείρισης βάσεων δεδομένων. Δημιουργήθηκε από την εταιρεία της Oracle και βασίζεται στην δομημένη σειρά εντολών SQL. Μέσω της σελίδας <http://localhost/phpmyadmin/> δίνεται ένα εύχρηστο περιβάλλον εργασίας για τη διαχείριση βάσεων.



Σχήμα 6: Διαχειριστικό Περιβάλλον - SQL μέσω PhpMyAdmin

2.2.1.3 FileZilla



Σχήμα 7: Λογότυπο - FileZilla

Το Filezilla πρόκεται για ένα πρόγραμμα το οποίο εκτελείται στη πλευρά του σέρβερ επιτρέποντας όποιον χρήστη (πελάτη) FTP, να συνδεθεί με σκοπό τη μεταφορά αρχείων, είτε αφορά το ανέβασμα είτε το κατέβασμα αυτών.

2.2.1.4 Mercury



Σχήμα 8: Λογότυπο - Mercury

Το πρόγραμμα Mercury, αποτελεί μέρος της εφαρμογής του Xampp, υλοποιώντας μια βασική λειτουργία, αυτή του τοπικού διαχειριστή αλληλογραφίας (mail server).

2.2.1.5 Tomcat



Σχήμα 9: Λογότυπο - Tomcat

Το πρόγραμμα Tomcat αποτελεί σέρβερ για αρχεία μορφής JavaServer στη περίπτωση που ο προγραμματιστής επιθυμεί τη συγγραφή κώδικα σε γλώσσα προγραμματισμού, JAVA .

2.2.2 HTML 5



Σχήμα 10: Λογότυπο - HTML 5

Η HTML είναι μια γλώσσα σήμανσης (και όχι προγραμματισμού). Το όνομα της πρέρχεται από τη σύντμηση των αρχικών γραμμάτων της, HyperText Markup Language,

όπου στα ελληνικά μεταφράζεται ως Γλώσσα Σήμανσης Υπερκειμένου. Είναι γλώσσα που περιγράφει τη δομή της ιστοσελίδας στο πλοηγητή.

Η χρήση της έκδοσης 5 (HTML 5), ξεκίνησε το 2008 και όταν εκδόθηκε είχε σκοπό τη παρουσίαση υλικού, χωρίς τη χρήση τρίτων προγραμμάτων, όπως επεκτάσεις. Ο λόγος ήταν για να μπορούν οι εφαρμογές να τρέχουν ομαλά στο πλοηγητή.

2.2.3 CSS



Σχήμα 11: Λογότυπο - CSS 3

Η CSS πρόκεται για μια γλώσσα, η οποία ευθύνεται για την εμφάνιση μιας ιστοσελίδας, η οποία είναι γραμμένη με τη γλώσσα σήμανσης HTML. Το όνομα της προέρχεται από τα αρχικά της γράμματα, Cascading Style Sheet.

2.2.3.1 Bootstrap



Σχήμα 12: Λογότυπο - Bootstrap

Το Bootstrap είναι μια δημοφιλή βιβλιοθήκη (Framework) του CSS. Πρωτεύον στόχος του αποτελεί, η απεικόνιση της ιστοσελίδας σε κινητά τηλέφωνα. Η νεότερη έκδοση της βιβλιοθήκης είναι η έκδοση 5. Παρέχει πλήθος έτοιμων υλοποιήσεων σχεδίασης.

2.2.4 PHP



Σχήμα 13: Λογότυπο - PHP

Η PHP είναι μια γλώσσα προγραμματισμού, που αφορά τη δημιουργία ιστοσελίδων με περιεχόμενο που μπορεί να αλλάζει δυναμικά. Απαιτείται η χρήση κάποιου διαχομιστή, για παράδειγμα ο Apache, για την εκτέλεση του κώδικα ώστε να είναι ορατά τα αποτελέσματα στο πλοηγητή, σε μορφή γλώσσας σήμανσης όπου γίνεται κατανοητή.

2.2.5 JavaScript



Σχήμα 14: Λογότυπο - JavaScript

Η JavaScript είναι μια γλώσσα προγραμματισμού και αποτελεί μέρος υλοποίησης μιας ιστοσελίδας. Η JavaScript σε σύγκριση με τη PHP εκτελείται στην πλευρά του πελάτη, δηλαδή στο πλοηγητή και δεν απαιτείται η χρήση κάποιου διαχομιστή.

2.2.5.1 Ajax

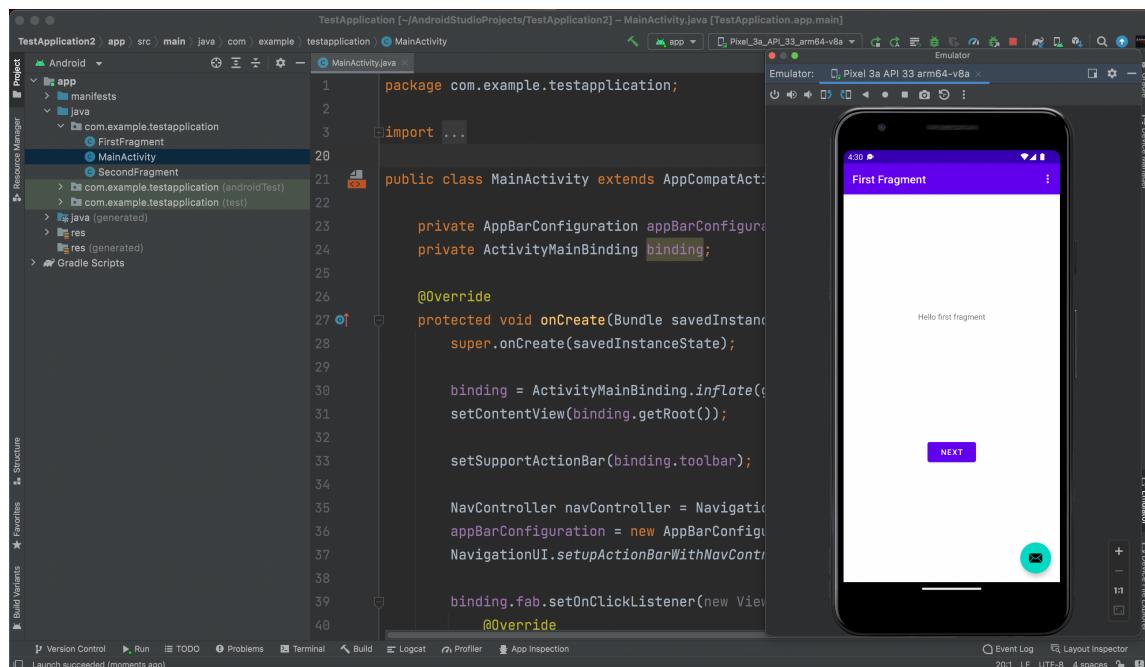


Σχήμα 15: Λογότυπο - AJAX

Ο AJAX είναι ένα σύνολο τεχνικών για τη πρόσβαση μιας ιστοσελίδας σε διακοσμητές (servers). Είναι μια μορφή ασύγχρονης JavaScript και XML εξού και το όνομα του, το οποίο είναι τα ακρωνύμια των λέξεων, **A**synchronous **J**ava**S**cript **A**nd **X**ML. Έχει την δυνατότητα αποστολής δεδομένων στο παρασκήνιο της ιστοσελίδας.

2.3 Android Studio

Το Android Studio, αποτελεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE), περιέχει τα απαραίτητα εργαλεία για την άναπτυξη λογισμικού, για συσκευές που έχουν ως λειτουργικό σύστημα, το Android.



Σχήμα 16: Περιβάλλον - Android Studio IDE

Κεφάλαιο 3

Τεχνολογίες Υλικού - Hardware

Σε αυτό το κεφάλαιο θα γίνει ανάλυση των υλικών που χρησιμοποιήθηκαν για τη κατασκευή της ηλεκτρονικής ζυγαριάς.

3.1 Arduino Uno Rev 3

Το Arduino πρόκειται για μια πλακέτα ανοικτού κώδικα βασισμένο στο μικροεπεξεργαστή, ATmega328P.

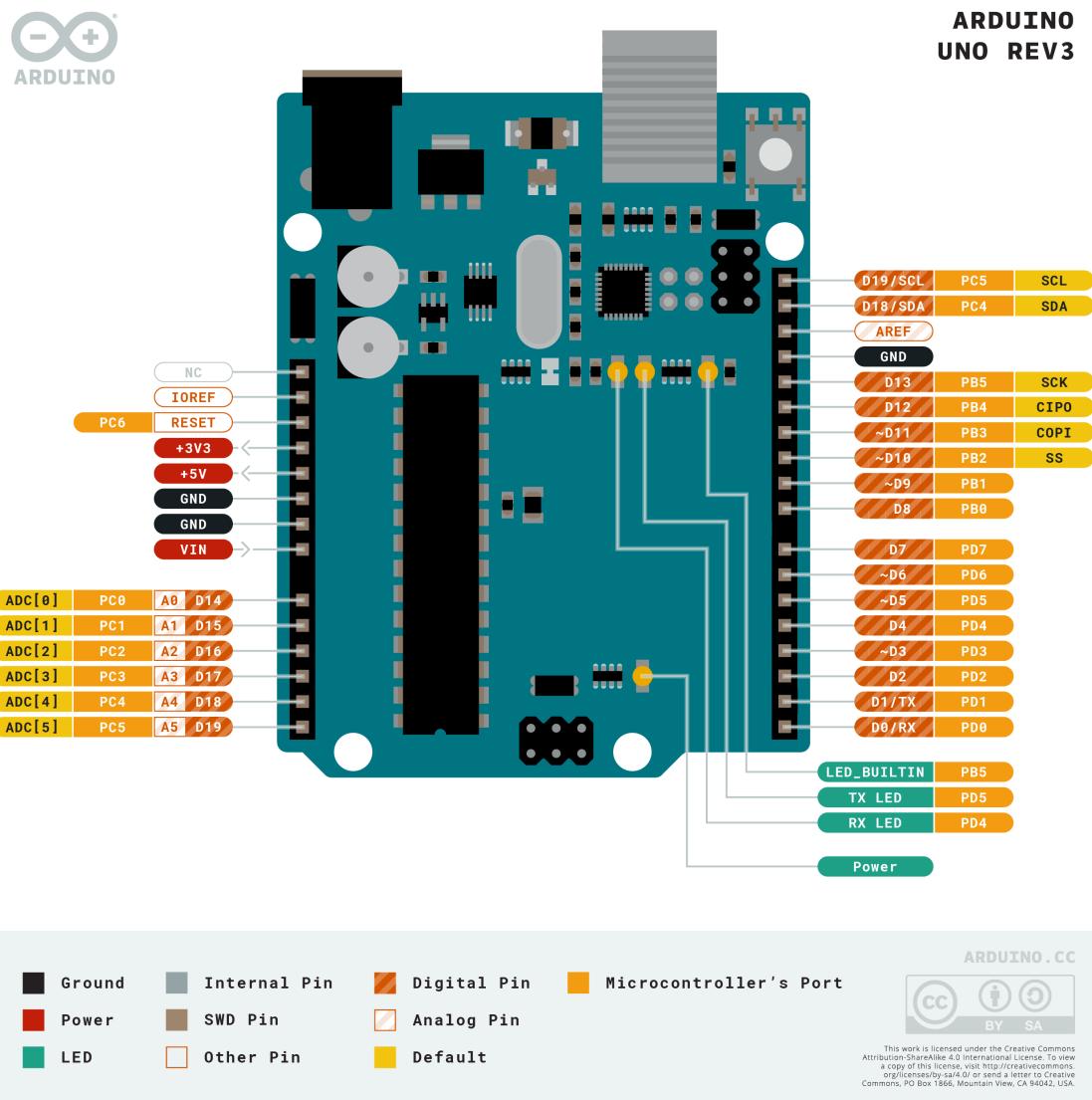


Σχήμα 17: Arduino Uno Rev 3

Χαρακτηριστικά:

Μικροελεγκτής:	ATmega328
Τάση λειτουργίας:	5 Volt
Τάση εισόδου:	7-12 Volt
Τάση εισόδου (limits):	6-20 Volt
Ψηφιακά I/O Pins:	14
Αναλογικές εισόδους:	6
PWM εισόδους:	6
DC Ρεύμα ανά I/O Pin:	20 mA
DC Ρεύμα για 3.3 Volt Pin:	50 mA
Mνήμη Flash:	32 KB
SRAM:	2 KB
EEPROM:	1 KB
Clock Speed:	16 MHz
Διαστάσεις Πλακέτας:	53.4 x 68.6 mm
Βάρος:	25g

Το παραχάτω διάγραμμα παρουσιάζει τη διάταξη των ακίδων (pins), του ολοκληρωμένου κυκλώματος (Arduino Uno) και τις λειτουργίες τους:



Σχήμα 18: Διάταξη ακίδων του Arduino Uno Rev 3

3.2 Arduino GSM Shield 2 (antenna connector)

Η επικοινωνία με το δίκτυο GSM (Global System for Mobile Communications), πραγματοποιείται μέσω της πλακέτας Arduino GSM Shield 2, η οποία παρέχει πρόσβαση στο Internet (GPRS). Ακόμη έχει τη δυνατότητα για αποστολή και λήψη γραπτών μηνυμάτων, SMS, αλλά και πραγματοποίηση κλήσεων ομιλίας μέσω του δικτύου 2G/3G (GPRS 850/900/1800/1900 MHz).

Οι δυνατότητες αυτές, προσφέρονται από τη βιβλιοθήκη GSM Library (<https://docs.arduino.cc/en/Reference/GSM>). Η πλακέτα, λειτουργεί και ως GSM modem.

Απαιτείται η χρήση κάρτας SIM. Η υποδοχή της κάρτας SIM είναι mini μεγέθους (2.5 * 1.5), ενώ άλλου μεγέθους απαιτείται η χρήση αντάπτορα. Για λόγους ασφαλείας, απαιτείται η χρήση κωδικού PIN όμως μπορεί να χρησιμοποιηθεί και χωρίς κωδικό PIN, αλλά δεν συνίσταται.

Τα βήματα για τη σύνδεση είναι τα εξής:

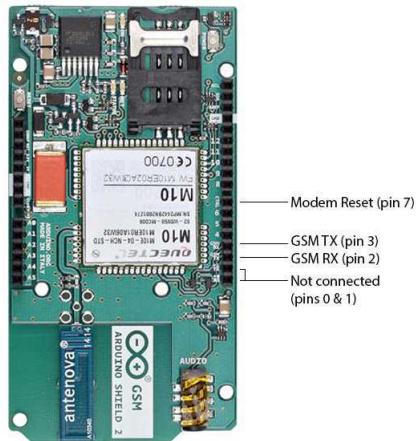
- Σύνδεση της πλακέτας επικοινωνίας (GSM) με τη πλακέτα Arduino (Uno, Mega) Η σύνδεση γίνεται εύκολα. Τα pins της πλακέτας GSM, εισέρχονται στις αντίστοιχες υποδοχές της πλακέτας του Arduino.
- Άνοιγμα του συρταριού της βάσης της κάρτας SIM και τοποθέτηση της κάρτας SIM εντός της υποδοχής.
- Κλείσιμο της υποδοχής με συρταρωτό τρόπο μέχρι να ασφαλίσει.

Οι υποδοχές που βρίσκονται στις θέσεις 2,3 και 7 είναι δεσμευμένες από το σύστημα για επικοινωνία μεταξύ των δύο πλακετών (Arduino Uno - GSM Shield). Υπάρχει και υποδοχή για την σύνδεση μικροφώνου για τη καταγραφή ήχου κατά τη διάρκεια της τηλεφωνικής κλήσης.



Σχήμα 19: GSM Shield 2

Διάγραμμα που δείχνει τη διάταξη των ακίδων της πλακέτας GSM:



Σχήμα 20: Διάταξη ακίδων της GSM Shield Antenna 2

3.3 Προέκταση κεραίας GSM

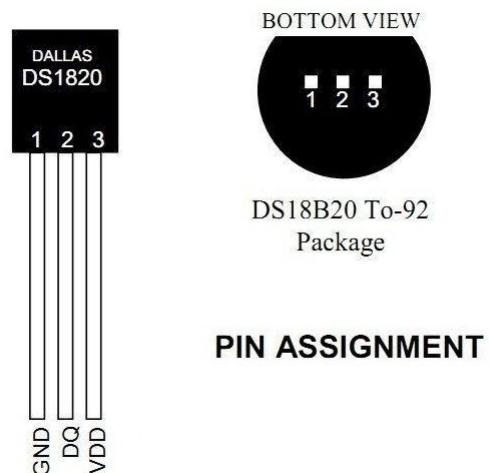
Για ενίσχυση του σήματος και εξασφάλιση καλύτερης ποιότητας, χρησιμοποιείται προέκταση 3m κεραίας GSM. Η εγκατάσταση της κεραίας, γίνεται στο ύψος της κυψέλης. Η σύνδεση γίνεται μέσω καλωδίου RP-SMA male/female.



Σχήμα 21: Κεραία GSM

3.4 Αισθητήρας Θερμοκρασίας DS18B20

Ο αισθητήρας της θερμοκρασίας, είναι ο DS18B20. Πρόκεται για ένα ψηφιακό αισθητήρα αδιάβροχο, με μεγάλη ακρίβεια στη μέτρηση θερμοκρασίας έχοντας $\pm 0.5^{\circ}\text{C}$ απόκλιση. Για τη σύνδεση με το μικροελεγκτή απαιτείται μια αντίσταση των 4.7Kohm.



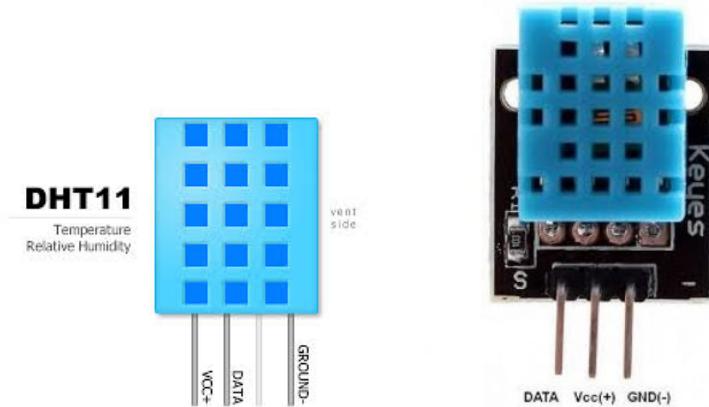
Σχήμα 22: Αισθητήρας Θερμοκρασίας DS18B20

Χαρακτηριστικά:

Κατασκευαστής:	Dallas Semiconductor
Όνομα Μοντέλου:	DS18B20
Τάση εισόδου (limits):	3-5.5 Volt
Εύρος Μετρήσεων Θερμοκρασίας:	-55 °C έως +125 °C
Διαστάσεις Υλικού:	7 - 9mm
Σύνδεση:	(1): GND (2): DATA (3): 3-5.5 Volt

3.5 Αισθητήρας Υγρασίας DHT11

Ο αισθητήρας υγρασίας είναι ο DHT11. Αποτελεί έναν από τους πιο οικονομικούς αισθητήρες, για τη μέτρηση της υγρασίας και της θερμοκρασίας. Στην αγορά κυκλοφορεί σε δύο εκδοχές, μια με ενσωματωμένη αντίσταση με 3 pins και μια χωρίς αντίσταση με 4 pins. Στην ηλεκτρονική ζυγαριά χρησιμοποιήθηκε, η έκδοση χωρίς αντίσταση.



Σχήμα 23: Αισθητήρας υγρασίας DHT11
 Αριστερά (4 pins - απαιτείται η χρήση αντίστασης)
 Δεξιά (3 pins - έτοιμη πλακέτα)

Χαρακτηριστικά:

Κατασκευαστής:	OEM
Όνομα Μοντέλου:	DHT11
Τάση εισόδου:	3-5.5 Volt
Εύρος Μετρήσεων Υγρασίας:	20% έως 80%
Εύρος Μετρήσεων Θερμοκρασίας:	-0 °C έως +50 °C με ακρίβεια ±2°C
Διαστάσεις Υλικού:	21.9 x 16.6 x 7mm
Κατανάλωση Ρεύματος:	2.5mA max

3.6 Αντίσταση 4.7 & 10 Kohm

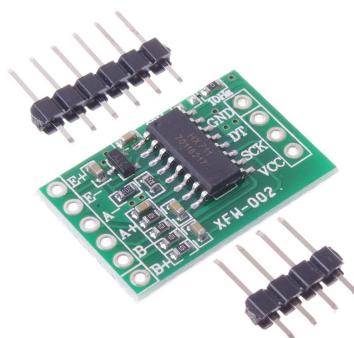
Απαιτείται για τη σωστή συνδεσμολογία του αισθητήρα της υγρασίας ($10\text{K}\Omega$) και της θερμοκρασίας ($4.7\text{K}\Omega$). Είναι κατασκευασμένη από άνιθρακα, δίνοντας έτσι τη δυνατότητα μεγαλύτερης αντοχής σε σήματα υψηλής ενέργειας.



Σχήμα 24: Resistor $4.7\text{ k}\Omega$ (στα αριστερά) και Resistor $10\text{ k}\Omega$ (στα δεξιά)

3.7 Μετατροπέας 24-bit A/D Conversion with HX711

Για τη καταγραφή βάρους από τον αισθητήρα, απαιτείται η χρήση ενός μετατροπέα, ο οποίος καταλαβαίνει τις αλλαγές στην αντίσταση του αισθητήρα βάρους. Εφόσον οριστεί ο calibration factor, όρος που χρησιμοποιείται για τη ρύθμιση της εναλλαγής της τιμής της αντίστασης. Αυτό επιτυγχάνεται με μετρήσεις σε αντικείμενο που το βάρος του είναι γνωστό.



Σχήμα 25: HX 711

3.8 Αισθητήρας βάρους αντοχής 150 kg

Οι αισθητήρες βάρους ανέρχονται σε πολλές εκδόσεις. Η επιλογή του αισθητήρα βάρους ανέρχεται στα 150 kg, το οποίο είναι μια μέγιστη τιμή που πιθανόν μπορεί να φτάσει μια χυψέλη. Δεν συνηθίζεται αυτό βάρος, καθώς δυσκολεύει αρκετά το μελισσοκόμο στη μεταφορά της χυψέλης.

Για την εξασφάλιση λειτουργίας του πρέπει το βάρος που καταμετρά να μην υπερβαίνει τα όρια αντοχής του. Με αυτή τη σκέψη, ορίστηκε και η μέγιστη αντοχή που απαιτεί το σύστημα.



Σχήμα 26: Weight Sensor 150 kg

Χαρακτηριστικά:

Ακρίβεια μέτρησης:	0.02
Σφάλμα σύνθεσης:	0.03
Τάση εισόδου:	9 - 12 Volt
Εύρος Μετρήσεων Θερμοκρασίας:	-35 °C έως +65 °C
Διαστάσεις Υλικού:	15 x 4 x 2,5 mm

3.9 Jumper Cables

Για τη σύνδεση των αισθητήρων με το Arduino, χρησιμοποιήθηκαν καλώδια τύπου jumper όλων των ειδών (female to male, male to male, female to male).



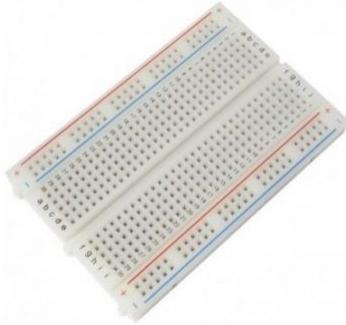
Σχήμα 27: Jumper Cables

3.10 Breadboard 400 οπών (Half size)

Πρόκειται για δοκιμαστική πλακέτα, στην οποία τοποθετήθηκαν οι αισθητήρες για δοκιμές μέχρι και τη πλήρη ανάπτυξη του κώδικα για το καλύτερο δυνατό έλεγχο του προγράμματος του Arduino.

Χρησιμοποιείται κυρίως για δημιουργία προτύπων κυκλωμάτων. Δεν απαιτείται η χρήση κολλητηριού. Υπάρχουν συνολικά 400 οπές. Οι 300 οπές χωρίζονται σε 10 στήλες και σε 30 γραμμές και αφορούν την τοποθέτηση εξαρτημάτων. Οι εναπομείναντες 100 οπές χρησιμοποιούνται για παροχή τροφοδοσίας.

Υπάρχουν και άλλα μοντέλα με λιγότερες ή περισσότερες οπές ανάλογα τις ανάγκες του κυκλώματος.



Σχήμα 28: Breadboard 400 θέσεων

Χαρακτηριστικά:

Υλικό κατασκευής:	Πλαστικό
Κατανάλωση ρεύματος:	300V/3-5A (current)
Διαστάσεις Υλικού:	83.5 x 54.5 x 8.5 mm

3.11 Μπαταρία 12 Volt 7.2Ah

Για τη τροφοδοσία του κυκλώματος, χρησιμοποιήθηκε μπαταρία των 12 Volt με χωρητικότητα, 7.2 Ah. Επιλέχθηκε λόγω της μεγάλης χωρητικότητας της και το μεγάλο προσδόκιμο ζωής. Συνήθως, χρησιμοποιείται σε συστήματα UPS, τηλεπικοινωνιακά συστήματα και συστήματα συναγερμού.



Σχήμα 29: Μπαταρία 12Volt 7.2Ah

3.12 Αντάπτορας για τροφοδότηση του Arduino

Για τη σύνδεση του Arduino με τη μπαταρία απαιτείται ειδικός αντάπτορας. Ο συγκεκριμένος αντάπτορας που παρουσιάζεται στην εικόνα αφορά τη σύνδεση της πλακέτας του Arduino με 9 Volt μπαταρία. Για τη σύνδεση του Arduino με τη μπαταρία των 12 Volt χόπηκε το βύσμα και ενώθηκε με καλώδιο.



Σχήμα 30: Αντάπτορας Μπαταρίας

3.13 Φωτοβολταϊκός Συλλέκτης 15W - 12V

Για την εξασφάλιση ενέργειας έγινε χρήση φωτοβολταικού συλλέκτη ενέργειας των 15 Watt και των 12 Volt.



Σχήμα 31: Φωτοβολταϊκός Συλλέκτης 15W - 12V

3.14 Ρυθμιστής Φόρτισης 20Ah 12Volt - 24Volt

Για τη φόρτιση της μπαταρίας από το φωτοβολταϊκό συλλέκτη, χρησιμοποιήθηκε ρυθμιστής φόρτισης όπου συνδέθηκε η μπαταρία και ο χρονοδιακόπτης με το Arduino.



Σχήμα 32: Ρυθμιστής Φόρτισης Μπαταρίας 20Α

3.15 Στεγανά Κουτιά

Για την αποθήκευση των εξαρτημάτων, κυκλωμάτων και την εξασφάλιση της άρτιας λειτουργία τους χρησιμοποιήθηκαν στεγανά κουτιά για μεγαλύτερη ασφάλεια. Για τη μεγιστοποίηση της στεγανοποίησης αυτών, χρησιμοποιήθηκαν υλικά μόνωσης.



Σχήμα 33: Κουτί Στεγανό

3.16 Κύκλωμα Timer Relay

Ο χρονοδιαχόπτης παρέχει τάση στο κύκλωμα του Arduino για συγκεκριμένο χρονικό διάστημα. Αυτό ορίζεται από το προγραμματιστή, ώστε να εξασφαλιστεί ο ημερήσιος αριθμός γραπτών μηνυμάτων (SMS) που θα στέλνονται από το σύστημα της ηλεκτρονικής ζυγαριάς.



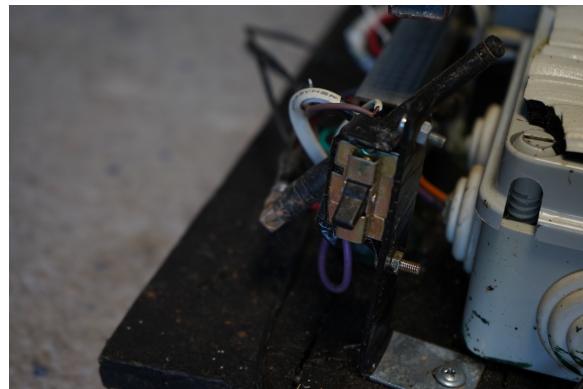
Σχήμα 34: Timer Relay

Χαρακτηριστικά:

Επιλογή Χρόνων Λειτουργίας:	0.1-99.9 sec, 1-999 sec, 1-999 min
Εύρος Λειτουργίας Θερμοκρασίας:	-40 °C έως +85 °C
Κατανάλωση ρεύματος:	20mA (current)
Κατανάλωση ρεύματος (Ανοικτό ρελέ):	50mA (current)
Διαστάσεις Υλικού:	64 x 35 x 18 mm
Βάρος:	25 g

3.17 Διακόπτες on/off

Για τη καλύτερη διαχείριση της συσκευής τοποθετήθηκε διακόπτης on/off για την άμεση επιλογή λετουργίας. Αφορά τη παροχή ρεύματος στο χρονοδιακόπτη.



Σχήμα 35: Διακόπτης on/off

3.18 Χειροποίητο κουτί κατασκευής

Για την αποφυγή εισόδου σκουπιδιών, υγρασίας αλλά και την αποφυγή εισβολής ζώων, όπως για παράδειγμα, φίδια ή άλλα επερτά, επικίνδυνα για τη κυψέλη, κατασκευάστηκε ειδικό κουτί με ξύλο θαλάσσης για τη προστασία του συστήματος.



Σχήμα 36: Χειροποιήτο κουτί αποθήκευσης ηλεκτρονικής ζυγαριάς

Παραχάτω, παρουσιάζεται η ηλεκτρονική ζυγαριά σε λειτουργία έχοντας τοποθετημένη μια κυψέλη και πάνω από αυτή το φωτοβολταϊκό πάνελ. Στο πίσω μέρος φαίνεται και το καλώδιο επέκτασης της κεραίας.



Σχήμα 37: Ηλεκτρονική Ζυγαριά - Τοποθετημένη Κυψέλη

3.19 Κοστολόγιο υλικών

Όλα τα απαραίτητα υλικά για τη κατασκευή της ηλεκτρονικής ζυγαριάς.¹

N/N	Προιόν	Ποσότητα	Τιμή
1	Arduino Uno Rev3	1	22.90 €
2	Καλώδιο Usb A to B	1	0.40 €
3	Breadboard	1	2.50 €
4	Jumper Male - Male Καλώδια	40	3.00 €
5	Αισθητήρας Θερμοκρασίας DS18B20	1	1.80 €
6	Αισθητήρας Υγρασίας DHT11	1	1.90 €
7	Πλακέτα ενίσχυσης σήματος HX711	1	2.50 €
8	Αισθητήρας Βάρους 150 kg	1	31.00 €
9	Jumper Male - Female Καλώδια	10	1.80 €
10	GSM Shield	1	119.00 €
11	Μπαταρία 12Volt 7.2 Ah	1	15.00 €
12	Διακόπτης	1	4.00 €
13	DC 12Volt Relay Module	1	10.00 €
14	Στεγανό Κουτί 10 x 10	1	12.00 €
15	Στεγανό Κουτί 3 x 3	1	8.00 €
16	Σιδηροκατασκευή για τον αισθητήρα βάρους	1	20.00 €
17	Ξύλα θαλάσσης (χειροποίητη κατασκευή)	1	15.00 €
18	Ηλιακό Πάνελ	1	12.35 €
19	Ρυθμιστής - Φορτιστής	1	12.30 €
20	Καλώδιο σύνδεσης Φορτιστή με Μπαταρία	3	12.00 €
21	Αντάπτορας jack για την τροφοδότηση του Arduino	1	0.73 €
22	Αντίσταση Carbon 1/4W 5% 4.7 Kohm	10	0.08 €
Σύνολο		80 τεμάχια	308.26 €

¹Οι τιμές είναι ενδεικτικές καθώς η αγορά τους πραγματοποιήθηκε το έτος 2018.

Κεφάλαιο 4

Σχεδιασμός του Συστήματος

4.1 Δεδομένα

Θα δημιουργηθεί ένα ολοκληρωμένο σύστημα απομακρυσμένης παρακολούθησης, το οποίο θα δίνει καθημερινά τις τιμές του βάρους, της θερμοκρασίας και της υγρασίας ενός μελισσού. Ο λόγος για τον οποίο γίνεται είναι για τη διαπίστωση της κατάστασης του περιβάλλοντος, αλλά και της πορείας του (μελισσού) στη παραγωγή αγαθών.

Με το παρόν σύστημα, θα δίνεται η δυνατότητα στο χρήστη - μελισσοκόμο να βλέπει σε άμεσο χρόνο, όλες τις προαναφερόμενες πληροφορίες που αφορούν το μελισσοκομικό του λαό. Αυτές οι δυνατότητες, θα προσφέρονται μέσω διαδικτυακής πλατφόρμας και μέσω τηλεματικής εφαρμογής σε συσκευές με λειτουργικό σύστημα, Android.

Σκοπός, η καλύτερη δυνατή παρακολούθηση από τους χρήστες για τη πρόληψη καταστάσεων και αναγκών του λαού του μελισσού, όπως για παράδειγμα η έλλειψη τροφής ή απότομη αύξηση του βάρους με σκοπό την άμεση παρέμβαση τους.

4.2 Ζητούμενα

4.2.1 Γενική περιγραφή λειτουργιών του συστήματος

1. Εγγραφή νέου χρήστη-μελισσοκόμου μέσω διαδικτυακής πλατφόρμας.
2. Εισαγωγή του χρήστη-μελισσοκόμου στη διαδικτυακή πλατφόρμα.
3. Εμφάνιση όλων των στοιχείων του προφίλ του χρήστη-μελισσοκόμου.
4. Εμφάνιση όλων των δεδομένων (υγρασία, θερμοκρασία και βάρος) για τον εκάστοτε χρήστη-μελισσοκόμο.
5. Εμφάνιση πληροφοριών για συγκεκριμένο χρονικό διάστημα (τρέχουσα ημέρα, χθεσινή ημέρα, εβδομάδα, συγκεκριμένου μήνα εντός έτους και τέλος συγκεκριμένης ημέρας βάση αναζήτησης από επιλογή) για κάθε χρήστη-μελισσοκόμο.

4.2.2 Κατηγορίες χρηστών

- Επισκέπτης διαδικτυακού ιστότοπου.
- Κάτοχος μελισσοκομικής ζυγαριάς.
- Διαχειριστής.
- Επισκέπτης.

4.2.3 Περιβάλλον λειτουργίας

Το μέρος της ηλεκτρονικής ζυγαριάς, αφορά στη συλλογή των μετρήσεων από τους αισθητήρες και την αποστολή γραπτού μηνύματος SMS όταν υλοποιηθεί με γλώσσα υψηλού επιπέδου, η οποία πλησιάζει όσο πιο κοντά στη γλώσσα χαμηλού επιπέδου, για τη συγγραφή κώδικα για το μικροεπεξεργαστή του Arduino, Atmega.

Το μέρος της τηλεματικής εφαρμογής, αφορά τη προβολή δεδομένων και την ανάγνωση του μηνύματος, με κατάλληλο τρόπο, από τη κινητή συσκευή. Αυτό έχει ως σκοπό, την εγγραφή των μετρήσεων στη βάση δεδομένων, με χρήση γλώσσας ανωτέρου επιπέδου. Η λειτουργία του, όταν καθίσταται δυνατή μέσω περιβάλλοντος κινητής συσκευής, με λειτουργικό σύστημα, Android.

Αντίστοιχο περιβάλλον προβολής δεδομένων - αποτελεσμάτων, όταν είναι υποστηριζόμενο από περιηγητή. Ακόμη όταν είναι υπεύθυνο, για την επικοινωνία της τηλεματικής εφαρμογής, με τη βάση δεδομένων, για τη σωστή ενημέρωση της αλλά και τη προβολή δεδομένων σε αυτήν.

4.2.4 Περιορισμοί στην σχεδίαση και την υλοποίηση

Για την εξασφάλιση της ομαλής λειτουργίας της ηλεκτρονικής ζυγαριάς, απαιτείται η σωστή διαχείριση της ενέργειας. Η τροφοδοσία γίνεται μέσω μπαταρίας. Ακόμη πρέπει να υπάρχει επαρκής κάλυψη σήματος, με σκοπό, τη βεβαιωμένη αποστολή γραπτού μηνύματος.

Για την επίλυση τυχόν προβλημάτων προτείνεται η χρήση ενός ηλιακού συλλέκτη και φορτιστή μπαταρίας. Επίσης, το πρόβλημα της κάλυψης σήματος μπορεί να καλυφθεί, με την εγκατάσταση προέκτασης κεραίας GSM, εκτός της κυψέλης.

Σε περίπτωση αδυναμίας κάλυψης σήματος ακόμη και μετά τη τοποθέτηση κεραίας λόγω της τοποθεσίας, προτείνεται η εγκατάσταση του συστήματος σε άλλο μέρος κοντά σε σχέση το μελισσοκομικό λαό.

4.3 Λειτουργίες Συστήματος

4.3.1 Πλοιήγηση επισκέπτη

Περιγραφή Λειτουργίας

Με την λειτουργία ο επισκέπτης μπορεί να πλοιηγηθεί στη σελίδα και να ενημερωθεί για τις δυνατότητες αλλά και τα υλικά της ηλεκτρονικής ζυγαριάς. Ακόμη, δίνεται η δυνατότητα επικοινωνίας με το διαχειριστή, μέσω αντίστοιχης φόρμας.

Ακολουθίες ερευθίσματος / απόκρισης

Ο χρήστης εισάγει τις απαιτούμενες πληροφορίες στα κατάλληλα πεδία της φόρμας επικοινωνίας και επιλέγει ‘Αποστολή’ για να στείλει μήνυμα (email) στο διαχειριστή. Σε περίπτωση που επιθυμεί να ακυρώσει τη διαδικασία μπορεί να επιστρέψει στην αρχική σελίδα.

Ανάλυση σε λειτουργικές απαιτήσεις

Δεν υπάρχουν λειτουργικές απαιτήσεις.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.2 Αλλαγή στοιχείων χρήστη - μελισσοκόμου στο σύστημα

Περιγραφή Λειτουργίας

Τηλεφωνικό περίπτωση ο μελισσοκόμος να αλλάξει πάροχο κινητής τηλεφωνίας, επομένως και τηλεφωνικό αριθμό ή επιμυμεί την αλλαγή του προσωπικού του κωδικού. Θα πρέπει να δίνεται στο χρήστη, η δυνατότητα αλλαγής των στοιχείων που είχε εισάγει κατά την εγγραφή του στο σύστημα.

Ακολουθίες ερευθίσματος / απόκρισης

Ο χρήστης, εισάγει το όνομα χρήστη (username) και το κωδικό πρόσβασης (password) και επιλέγει ‘Είσοδος’. Στη συνέχεια, επιλέγει ‘Προφίλ’ και πραγματοποιεί τις αλλαγές που επιθυμεί. Επιλέγει ‘Αποθήκευση Αλλαγών’ για την αποθήκευση των αλλαγών που έκανε. Σε περίπτωση επιθυμίας επιστροφής σε προηγούμενα στοιχεία μπορεί να γίνει η διαδικασία ξανά από την αρχή.

Ανάλυση σε λειτουργικές απαιτήσεις

Ισχύουν οι ίδιες με τις λειτουργίες 1 και 2.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.3 Εγγραφή νέου χρήστη - μελισσοκόμου

Περιγραφή Λειτουργίας

Με τη λειτουργία αυτή, ο εκάστοτε μελισσοκόμος, θα μπορεί να εισάγει και να καταχωρίσει τα απαριθμητά στοιχεία, στο σύστημα.

- Όνομα
- Επώνυμο
- Όνομα Χρήστη (username)
- Email
- Κωδικός Πρόσβασης
- Αριθμός κινητής τηλεφωνίας (ηλεκτρονικής ζυγαριάς)
- Αποδοχή όρων χρήσης

Ακολουθίες ερευθίσματος / απόκρισης

Ο χρήστης εισάγει τα δεδομένα στο αντίστοιχο πεδίο και μόλις τα συμπληρώσει όλα, επιλέγει "Εγγραφή". Σε περίπτωση, ο χρήστης επιθυμεί να ακυρώσει τη διαδικασία εγγραφής του στο σύστημα λίγο πριν πατήσει την 'Εγγραφή' απλά χρειάζεται η ανανέωση της σελίδας. Σε αντίθετη περίπτωση, θα πρέπει να επικοινωνήσει με το διαχειριστή.

Ανάλυση σε λειτουργικές απαιτήσεις

REQ-1: Όλα τα πεδία θα πρέπει να συμπληρωθούν, δηλαδή δεν θα πρέπει να υπάρχει κενό πεδίο.

REQ-2: Το όνομα χρήστη (username) και ο κωδικός πρόσβασης θα πρέπει να έχουν μόνο λατινικούς χαρακτήρες συμπεριλαμβάνοντας κεφαλαία.

REQ-3: Το όνομα χρήστη (username) θα πρέπει να είναι μοναδικό για κάθε χρήστη.

REQ-4: Το email κάθε χρήστη θα πρέπει να είναι μοναδικό μεταξύ όλων των χρηστών.

REQ-5: Ο χρήστης θα πρέπει να αποδέχεται τους όρους χρήσης του συστήματος.

REQ-6: Ο αριθμός κινητής τηλεφωνίας που αντιστοιχίζεται στην ηλεκτρονική ζυγαριά θα πρέπει να είναι μοναδικός για κάθε χρήστη.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.4 Εισαγωγή χρήστη - μελισσοκόμου στο σύστημα

Περιγραφή Λειτουργίας

Με την λειτουργία εισαγωγής στο σύστημα, ο εκάστοτε μελισσοκόμος θα μπορεί να αποκτήσει πρόσβαση στα στοιχεία του στο σύστημα, δίνοντας το όνομα χρήστη (username) και το κωδικό πρόσβασης.

Ακολουθίες ερευθίσματος / απόκρισης

Ο χρήστης εισάγει το όνομα χρήστη (username) και το κωδικό πρόσβασης και στη συνέχεια, επιλέγει 'Είσοδος'. Σε περίπτωση που θέλει να ακυρώσει τη διαδικασία επιλέγει 'Αποσύνδεση'.

Ανάλυση σε λειτουργικές απαιτήσεις

REQ-1: Τα στοιχεία που συμπληρώνει ο χρήστης θα πρέπει να επαληθεύονται από λογισμικό με σκοπό την διαφύλαξη ανεπιθύμητης πρόσβασης στο σύστημα.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.5 Αλλαγή στοιχείων χρήστη - μελισσοκόμου στο σύστημα

Περιγραφή Λειτουργίας

Πολλές φορές είναι δυνατόν κάποιος μελισσοκόμος να αλλάξει πάροχο κινητής τηλεφωνίας, άρα και αριθμό ή επιλυμένη την αλλαγή του προσωπικού του κωδικού. Θα πρέπει να δίνεται στον χρήστη η δυνατότητα αλλαγής των στοιχείων που είχε εισάγει κατά την εγγραφή του στο σύστημα.

Ακολουθίες ερευθίσματος / απόκρισης

Ο χρήστης εισάγει το όνομα χρήστη (username) και τον κωδικό πρόσβασης και επιλέγει 'Είσοδος'. Κατόπιν επιλέγει 'Προφίλ' και πραγματοποιεί τις αλλαγές που επιλυμένει και επιλέγει 'Αποθήκευση Αλλαγών'. Σε περίπτωση επιλυμίας επιστροφής σε προηγούμενα στοιχεία μπορεί να γίνει η διαδικασία ξανά από την αρχή.

Ανάλυση σε λειτουργικές απαιτήσεις

Ισχύοι οι ίδιες με τις λειτουργίες 1 και 2.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.6 Προβολή δεδομένων από την ηλεκτρονική ζυγαριά

Περιγραφή Λειτουργίας

Σε αυτή την λειτουργία ο μελισσοκόμος έχει την δυνατότητα προβολής των δεδομένων, θερμοκρασία, βάρος και υγρασία, από την ηλεκτρονική ζυγαριά είτε σε περιηγητή είτε σε κινητή συσκευή με λειτουργικό σύστημα Android.

Ακολουθίες ερευνώσματος / απόκρισης

Ο χρήστης εισάγει το όνομα χρήστη (username) και τον κωδικό πρόσβασης και επιλέγει 'Είσοδος'. Κατόπιν επιλέγει 'Διαχείριση' και παρακολουθεί τις μετρήσεις. Μπορεί να επιλέξει ανάμεσα σε επιλογές όπως είναι η σημερινή ημέρα, η χθεσινή, η εβδομάδα που διανύει, έναν συγκεκριμένο μήνα που επιθυμεί να δει συγκεντρωτικά άλλα ακόμη και να αναζητήσει συγκεκριμένης ημέρας, δεδομένα. Στην περίπτωση που ο χρήστης επιθυμεί την διακοπή της λειτουργίας επιλέγει 'Αποσύνδεση'.

Ανάλυση σε λειτουργικές απαιτήσεις

REQ-1: Ο χρήστης μπορεί να επιλέξει συγκεκριμένη ημέρα η οποία δεν είναι μελλοντική ή παρελθοντική από το 1970.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.7 Λειτουργίες διαχειριστή

Περιγραφή Λειτουργίας

- Προγραμματισμός ρυθμού αποστολής γραπτών μηνυμάτων
- Προγραμματισμός ρυθμίσεων σύνδεσης τηλεματικής εφαρμογής με την βάση δεδομένων

Ακολουθίες ερευθίσματος / απόκρισης

Ο διαχειριστής έχει την δυνατότητα να ελέγχει τον ρυθμό αποστολής γραπτών μηνυμάτων, ανάλογα με τις ανάγκες - απαιτήσεις του μελισσοκόμου. Μπορεί να οριστεί όσες φορές χρειάζεται με μέγιστο της 16 ώρες. Συνηθίζεται η ενημέρωση να γίνεται κάθε 12 ώρες ώστε ο χρήστης να λαμβάνει 2 γραπτά μηνύματα την ημέρα.

Ο διαχειριστής ελέγχει τις ρυθμίσεις σύνδεσης της τηλεματικής εφαρμογής με την βάση δεδομένων.

Ανάλυση σε λειτουργικές απαιτήσεις

Δεν υπάρχουν λειτουργικές απαιτήσεις.

Ανάλυση σε μη λειτουργικές απαιτήσεις

Δεν υπάρχουν μη λειτουργικές απαιτήσεις.

4.3.8 Καταγραφή Περιπτώσεων Χρήσης

4.3.8.1 Περίπτωση Χρήσης #1

Κωδικός Περίπτωσης:	UC-1	
Όνομασία:	Εγγραφή νέου χρήστη	
Δημιουργήθηκε από:	Κωνσταντίνος Σωφρονάς	Τελευταία ενημέρωση από:
Ημερομηνία Συγγραφής:	28/01/2022	Ημερομηνία τελευταίας ενημέρωσης:

Εμπλεκόμενοι Ρόλοι:	Κάτοχος ηλεκτρονικής ζυγαριάς.
Περιγραφή:	Εισαγωγή και καταχώριση δεδομένων του χρήστη στο σύστημα.
Γεγονός Εκκίνησης:	Επιλογή Sign In για εγγραφή από την αρχική φόρμα.
Προϋποθέσεις:	Η ιστοσελίδα να είναι online.
Τελική Κατάσταση:	Τα στοιχεία του μελισσοκόμου - χρήστη να καταγραφούν στην βάση δεδομένων.
Φυσιολογική Ροή:	<p>1.1 Ο χρήστης επιλέγει 'Sign In' από την αρχική σελίδα.</p> <p>1.2 Εισάγει τα στοιχεία του στα αντίστοιχα πεδία - κελιά.</p> <p>1.3 Εφόσον έχουν γίνει εισαγωγή τα δεδομένα, επιλέγει 'Εγγραφή'.</p> <p>1.4 Τα στοιχεία αποθηκεύονται στην βάση δεδομένων του συστήματος.</p>
Εναλλακτική Ροή:	<p>1.3 Ο χρήστης επιλέγει επιστροφή.</p> <p>1.4 Εμφάνιση αρχικής σελίδας.</p>
Εξαιρέσεις:	<p>1.4 Το σύστημα να βρίσκεται σε κατάσταση εκτός λειτουργίας.</p> <p>1.5 Εμφάνιση αρχικής σελίδας.</p>
Ενσωματώνει:	-
Προτεραιότητα:	Πολύ υψηλή.

Ειδικές απαιτήσεις:	<ol style="list-style-type: none">1 Όλα τα πεδία - κελιά της φόρμας θα πρέπει να έχουν συμπληρωμένα στοιχεία, δηλαδή να μην υπάρχουν πεδία κενά.2 Το όνομα του χρήστη θα πρέπει να είναι μοναδικό.3 Το email του χρήστη θα πρέπει να είναι μοναδικό.
Τυποθέσεις:	-

4.3.8.2 Περίπτωση Χρήσης #2

Κωδικός Περίπτωσης:	UC-2	
Όνομασία:	Είσοδος χρήστη στο σύστημα	
Δημιουργήθηκε από:	Κωνσταντίνος Σωφρονάς	Τελευταία ενημέρωση από:
Ημερομηνία Συγγραφής:	29/01/2022	Ημερομηνία τελευταίας ενημέρωσης:

Εμπλεκόμενοι Ρόλοι:	Κάτοχος ηλεκτρονικής ζυγαριάς.
Περιγραφή:	Ο χρήστης εισάγει το όνομα χρήστη (username) και τον κωδικό πρόσβασης και εισέρχεται στο σύστημα.
Γεγονός Εκκίνησης:	Επιλογή Log In από την αρχική φόρμα.
Προϋποθέσεις:	Η ιστοσελίδα να είναι online.
Τελική Κατάσταση:	Προβολή αρχικής σελίδας με την προσθήκη επιλογών στον χρήστη για την προβολή δεδομένων μέσω της επιλογής ‘Διαχείριση’.
Φυσιολογική Ροή:	<p>2.1 Ο χρήστης επιλέγει ‘Log In’ από την αρχική σελίδα.</p> <p>2.2 Εισάγει το όνομα χρήστη.</p> <p>2.3 Εισάγει το κωδικό πρόσβασης.</p> <p>2.4 Επιλέγει ‘Είσοδος’.</p> <p>2.5 Τα στοιχεία, όνομα και κωδικός πρόσβασης, στέλνονται στον διαχειριστή του συστήματος.</p> <p>2.6 Γίνεται έλεγχος των στοιχείων για την ορθότητα τους.</p> <p>2.7 Εμφάνιση αρχικής φόρμας με την προσθήκη επιπλεόν επιλογών όπως για την προβολή δεδομένων.</p> <p>2.8 Επιλέγει ‘Διαχείριση’ και εμφανίζεται η φόρμα προβολής δεδομένων.</p>
Εναλλακτική Ροή:	<p>2.3 Ο χρήστης επιλέγει επιστροφή στην αρχική σελίδα.</p> <p>2.4 Εμφάνιση αρχικής σελίδας.</p>

Εξαιρέσεις:	2.2 Το σύστημα να βρίσκεται σε κατάσταση εκτός λειτουργίας. 2.3 Εμφάνιση αρχικής σελίδας.
Ενσωματώνει:	-
Προτεραιότητα:	Πολύ υψηλή.
Ειδικές απαιτήσεις:	1 Τα στοιχεία περνούν από επαλήθευση από λογισμικό του διαχειριστή για την εξασφάλιση της ασφάλειας των δεδομένων και την αποφύγη οποιουδήποτε την είσοδο στο σύστημα.
Τυποθέσεις:	-

4.3.8.3 Περίπτωση Χρήσης #3

Κωδικός Περίπτωσης:	UC-3		
Ονομασία:	Αλλαγή στοιχείων χρήστη		
Δημιουργήθηκε από:	Κωνσταντίνος Σωφρονάς	Τελευταία ενημέρωση από:	
Ημερομηνία Συγγραφής:	29/01/2022	Ημερομηνία τελευταίας ενημέρωσης:	

Εμπλεκόμενοι Ρόλοι:	Κάτοχος ηλεκτρονικής ζυγαριάς.
Περιγραφή:	Ο χρήστης έχει την δυνατότητα αλλαγής των στοιχείων που είχε δηλώσει κατά την διάρκεια της εγγραφής του στο σύστημα.
Γεγονός Εκκίνησης:	Επιλογή 'Προφίλ' από την αρχική φόρμα.
Προϋποθέσεις:	Η ιστοσελίδα να είναι online.
Τελική Κατάσταση:	Τροποποίηση στοιχείων χρήστη και ενημέρωση βάσης δεδομένων με τις αλλαγές.

<p>Φυσιολογική Ροή:</p>	<ul style="list-style-type: none"> 3.1 Ο χρήστης επιλέγει 'Log In' από την αρχική σελίδα. 3.2 Εισάγει το όνομα χρήστη. 3.3 Εισάγει το κωδικό πρόσβασης. 3.4 Επιλέγει 'Είσοδος'. 3.5 Τα στοιχεία, όνομα και κωδικός πρόσβασης, στέλνονται στον διαχειριστή του συστήματος. 3.6 Γίνεται έλεγχος των στοιχείων για την ορθότητα τους. 3.7 Εμφάνιση αρχικής φόρμας με την προσθήκη επιπλεόν επιλογών όπως για την προβολή δεδομένων και την επεξεργασία στοιχείων του χρήστη. 3.8 Επιλέγει 'Προφίλ' και εμφανίζεται η φόρμα επεξεργασίας στοιχείων. 3.9 Ο διαχειριστής επικοινωνεί με την βάση δεδομένων. 3.10 Ο διαχειριστής στέλνει τα δεδομένα από την βάση δεδομένων στην φόρμα στοιχείων. 3.11 Ο χρήστης τροποποιεί, επεξεργάζεται, τα στοιχεία που επιθυμεί. 3.12 Εφόσον τελειώσει με τις αλλαγές επιλέγει 'Αποθήκευση Αλλαγών'. 3.13 Τα στοιχεία ελέγχονται από τον διαχειριστή του συστήματος με αλγόριθμο για την αποφύγη χρήσης στοιχείων από άλλον χρήστη, όπως το κινητό τηλέφωνο, το όνομα χρήστη και το email . 3.14 Εφόσον δεν υπάρχουν διπλοεγγραφές, τα στοιχεία αποθηκεύονται στην βάση δεδομένων και γίνεται ενημέρωση στοιχείων του \$_SESSION .
--------------------------------	--

Εναλλακτική Ροή:	3.3 Ο χρήστης επιλέγει επιστροφή στην αρχική σελίδα. 3.4 Εμφάνιση αρχικής σελίδας.
Εξαιρέσεις:	3.2 Το σύστημα να βρίσκεται σε κατάσταση εκτός λειτουργίας. 3.3 Εμφάνιση αρχικής σελίδας.
Ενσωματώνει:	-
Προτεραιότητα:	Μέτρια.
Ειδικές απαιτήσεις:	1 Τα στοιχεία περνούν από επαλήθευση από λογισμικό του διαχειριστή για την εξασφάλιση της ασφάλειας των δεδομένων και την αποφυγή οποιουδήποτε την είσοδο στο σύστημα.
Τυποθέσεις:	-

4.3.8.4 Περίπτωση Χρήσης #4

Κωδικός Περίπτωσης:	UC-4		
Ονομασία:	Προβολή Δεδομένων		
Δημιουργήθηκε από:	Κωνσταντίνος Σωφρονάς	Τελευταία ενημέρωση από:	
Ημερομηνία Συγγραφής:	29/08/2022	Ημερομηνία τελευταίας ενημέρωσης:	

Εμπλεκόμενοι Ρόλοι:	Κάτοχος ηλεκτρονικής ζυγαριάς.
Περιγραφή:	Ο χρήστης έχει την δυνατότητα προβολής των στοιχείων από την ηλεκτρονική ζυγαριά που έχουν εγγραφεί στο σύστημα.
Γεγονός Εκκίνησης:	Επιλογή 'Διαχείριση' από την αρχική φόρμα.
Προϋποθέσεις:	Η ιστοσελίδα να είναι online.
Τελική Κατάσταση:	Ενημέρωση του χρήστη για τις μετρήσεις από την ηλεκτρονική ζυγαριά.

Φυσιολογική Ροή:	<p>4.1 Ο χρήστης επιλέγει 'Log In' από την αρχική σελίδα.</p> <p>4.2 Εισάγει το όνομα χρήστη.</p> <p>4.3 Εισάγει το κωδικό πρόσβασης.</p> <p>4.4 Επιλέγει 'Είσοδος'.</p> <p>4.5 Τα στοιχεία, όνομα και κωδικός πρόσβασης, στέλνονται στον διαχειριστή του συστήματος.</p> <p>4.6 Γίνεται έλεγχος των στοιχείων για την ορθότητα τους.</p> <p>4.7 Εμφάνιση αρχικής φόρμας με την προσθήκη επιπλεόν επιλογών όπως για την προβολή δεδομένων και την επεξεργασία στοιχείων του χρήστη.</p> <p>4.8 Επιλέγει 'Διαχείριση' και εμφανίζεται η σελίδα επιλογής εμφάνισης μετρήσεων.</p> <p>4.9 Ο διαχειριστής επικοινωνεί με την βάση δεδομένων.</p> <p>4.10 Ο διαχειριστής στέλνει τα δεδομένα από την βάση δεδομένων στην προβολή στοιχείων.</p> <p>4.11 Ο χρήστης ενημερώνεται για τις μετρήσεις.</p>
Εναλλακτική Ροή:	<p>4.3 Ο χρήστης επιλέγει επιστροφή στην αρχική σελίδα.</p> <p>4.4 Εμφάνιση αρχικής σελίδας.</p>
Εξαιρέσεις:	<p>4.2 Το σύστημα να βρίσκεται σε κατάσταση εκτός λειτουργίας.</p> <p>4.3 Εμφάνιση αρχικής σελίδας.</p>
Ενσωματώνεις:	-
Προτεραιότητα:	Πολύ υψηλή.

Ειδικές απαιτήσεις:	1 Ο χρήστης μπορεί να επιλέξει για τις μετρήσεις επιλογές μεταξύ της σημερινής ημέρας, της χθεσινής, της εβδομάδας, κάποιου συγκεκριμένου μήνα εντός του έτους αλλά και κάποιας συγκεκριμένης ημέρας.
Τυποθέσεις:	Έχει γίνει σύνδεση του χρήστη με το σύστημα.

Κεφάλαιο 5

Υλοποίηση του Συστήματος

5.1 Arduino

Για το προγραμματισμό του Arduino Uno, υλοποιήθηκαν αρχεία δοκιμών με σκοπό τη διαπίστωση λειτουργίας του εκάστοτε αισθητήρα. Στη συνέχεια, έγινε η συγγραφή ενός ενιαίου προγράμματος συμπεριλαμβανόμενων όλων των αισθητήρων.

5.1.1 Απαραίτητες Βιβλιοθήκες

Για τη σύνδεση των αισθητήρων με το Arduino, είναι απαραίτητη η λήψη των απαιτούμενων βιβλιοθηκών για την ομαλή λειτουργία τους. Στο παρακάτω υπερσύνδεσμο υπάρχουν οι βιβλιοθήκες για τους περισσότερους αισθητήρες <https://www.arduino.cc/reference/en/libraries/category/sensors/> που χρησιμοποιούνται.

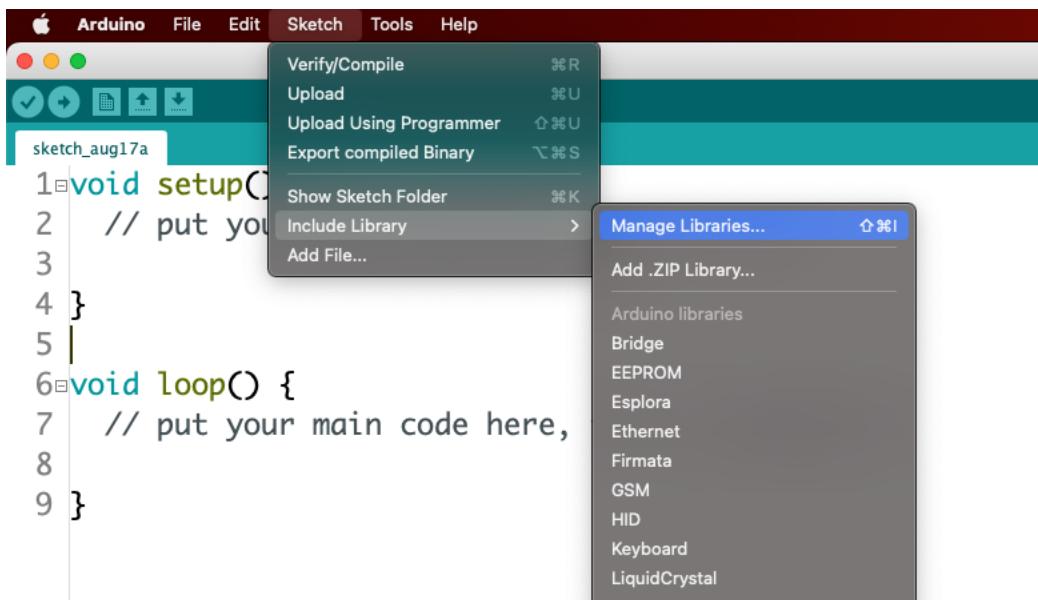
5.1.2 Τρόπος Εγκατάστασης Βιβλιοθηκών

Τπάρχουν δύο τρόποι εγκατάστασης μιας βιβλιοθήκης.

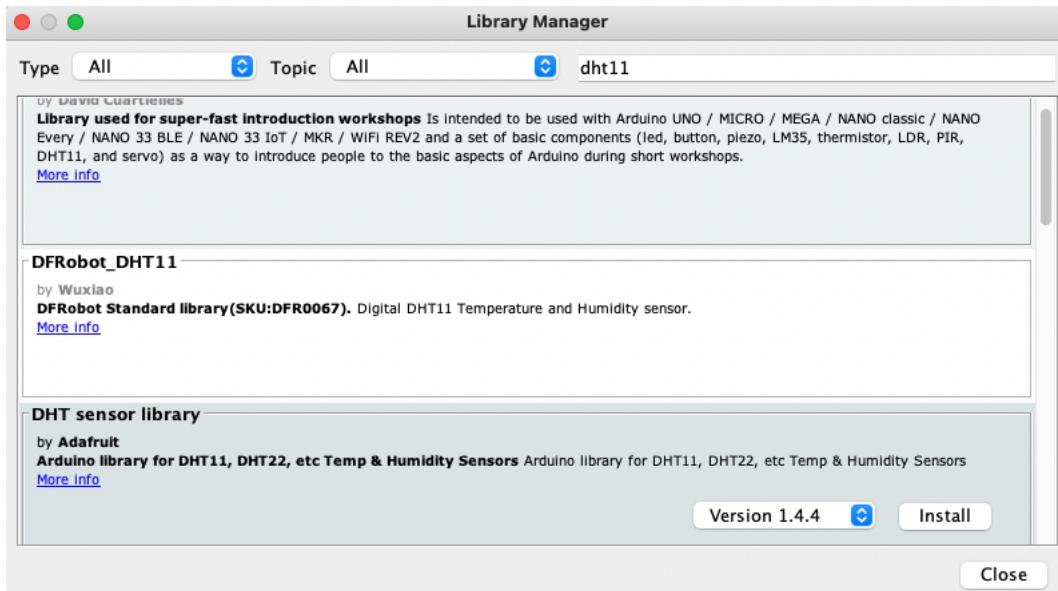
- Αυτόματος Τρόπος Εγκατάστασης Βιβλιοθήκης

Από το πρόγραμμα του Arduino IDE, στη καρτέλα (**Σχέδιο - Sketch**) από το αναδυόμενο μενού, επιλέγετε (**Συμπερίληψη Βιβλιοθήκης - Include Library**) και (**Διαχείριση Βιβλιοθηκών - Manage Library**). Στο παράθυρο που εμφανίζεται, με τη γραμμή αναζήτησης, κάνετε επιλογή στη βιβλιοθήκη της επιλογής σας για (**Εγκατάσταση**).

Σε κάποιες περιπτώσεις ο χρήστης μπορεί να επιλέξει και την έκδοση της βιβλιοθήκης. Εξαρτάται από την εκάστοτε βιβλιοθήκη αν έχει παλαιότερες εκδόσεις ή όχι. Ο σκοπός μιας παλαιότερης έκδοσης είναι η έλλειψη δυνατοτήτων ή ασυμβατότητα της έκδοσης με τον αισθητήρα, λόγω ύπαρξης προβλήματος που δεν έχει λυθεί ακόμη από το δημιουργό.



Σχήμα 38: Βήμα 1ο - Εγκατάσταση Βιβλιοθήκης - Arduino



Σχήμα 39: Βήμα 2ο - Αναζήτηση και Επιλογή Έκδοσης Βιβλιοθήκης - Arduino

- Χειροκίνητος Τρόπος Εγκατάστασης Βιβλιοθήκης

Ο χειροκίνητος τρόπος εγκατάστασης των βιβλιοθηκών, απαιτεί να αντιγράψετε τα αρχεία που λήφθηκαν, στην διαδρομή των βιβλιοθηκών. Αυτές συνήθως βρίσκονται στα Έγγραφα με ονομασία Arduino Libraries.

Τοποθεσία Βιβλιοθηκών για εκάστοτε λειτουργικό σύστημα:

Windows

C:\Users\username\Documents\Arduino\Libraries

Linux

/usr/bin/arduino/libraries

MacOS

/Users/username/Documents/arduino/libraries

5.1.3 Βιβλιοθήκες

Για τον αισθητήρα της θερμοκρασίας DS18B20:

- DS18B20:
<https://github.com/milesburton/Arduino-Temperature-Control-Library>
- One - Wire bus:
<https://github.com/PaulStoffregen/OneWire>

Για τον αισθητήρα της υγρασίας DHT11:

- DHT11:
<https://github.com/adidax/dht11>

Για τον αισθητήρα βάρους - και το μετατροπέα HX711:

- HX711:
<https://www.arduino.cc/reference/en/libraries/hx711-arduino-library>

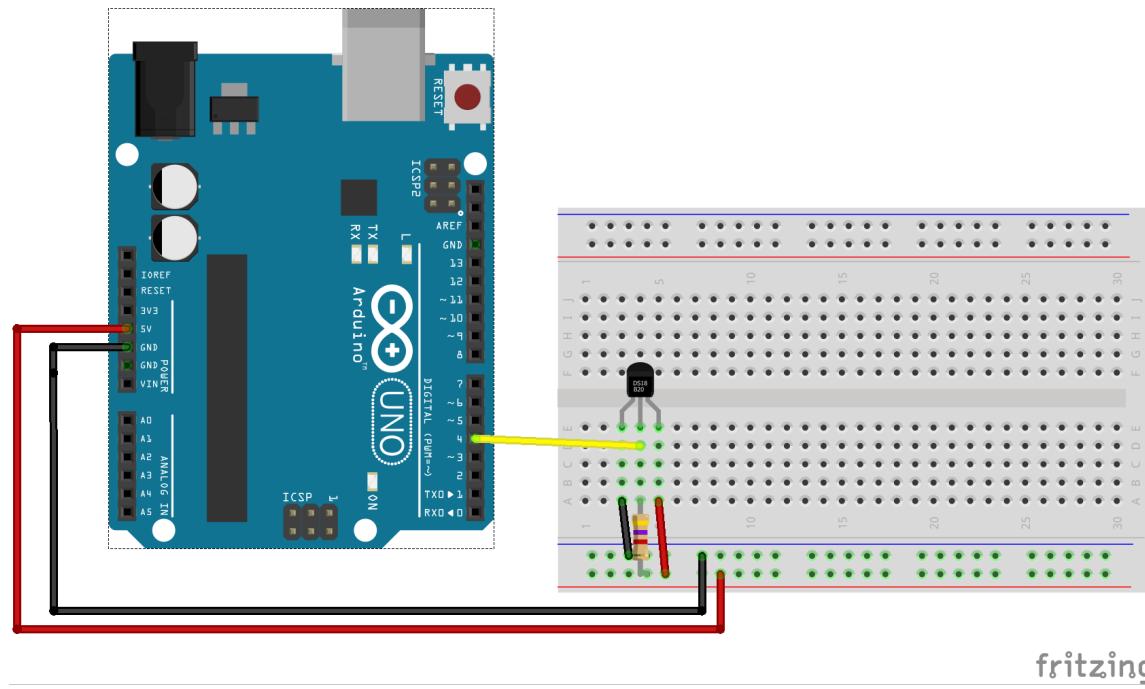
Για τη πλακέτα GSM Shield:

- GSM Shield Library:
<https://github.com/arduino-libraries/GSM>

5.1.4 Συνδεσμολογία Αισθητήρων με το Arduino

Ακολουθούν αναλυτικές οδηγίες σύνδεσης των αισθητήρων, με τη πλακέτα Arduino με χρήση πλακέτας δοκιμών (Breadboard).

5.1.4.1 Συνδεσμολογία DS18B20



Σχήμα 40: Συνδεσμολογία: DS18B20 - Arduino

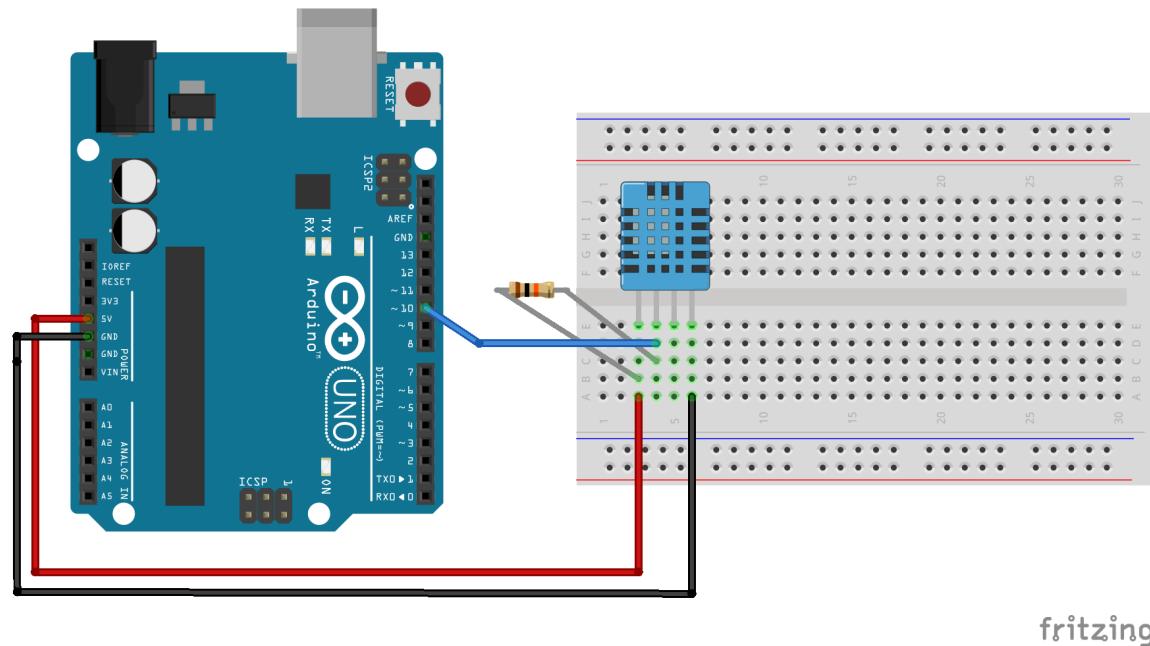
Κατά τη συνδεσμολογία του αισθητήρα χρειάζεται ιδιαίτερη προσοχή, διότι σε περίπτωση λανθασμένης σύνδεσης υπάρχει κίνδυνος καταστροφής του.

Απαιτείται αντίσταση 4.7 KΩ μεταξύ των 5 Volt και του pin (μεσσαίου) του αισθητήρα για τη λήψη δεδομένων.

5.1.4.2 Συνδεσμολογία DHT11

Ο αισθητήρας υγρασίας, κυκλοφορεί στο εμπόριο σε δύο εκδόσεις. Η μία έκδοση, συμπεριλαμβάνει στο κύκλωμα της, την αντίσταση (3 pin). Αντίθετα, η άλλη έκδοση (4 pin) χρήζει εξωτερικής αντίστασης μεγέθους 10 KΩ. Ακολούθουν αναλυτικά σχεδιαγράμματα συνδεσμολογίας και για τις δύο εκδόσεις:

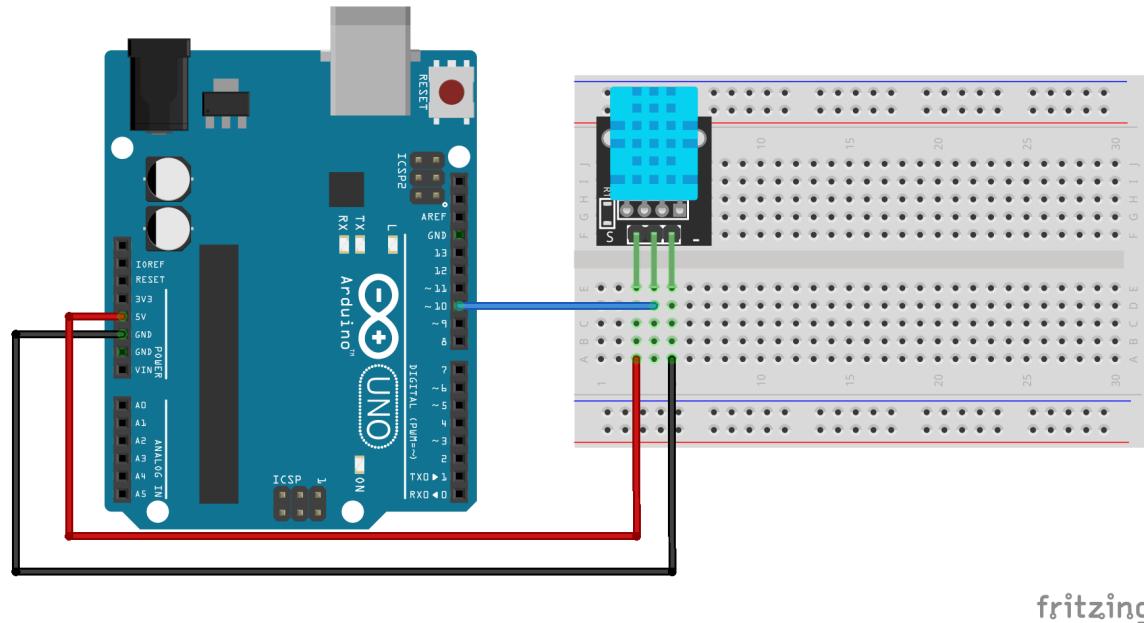
- 1η έκδοση: Σύνδεση αισθητήρα με 4 pin



Σχήμα 41: Συνδεσμολογία: DHT11 (4 pin) - Arduino

- 2η έκδοση: Σύνδεση αισθητήρα με 3 pin

Σε περίπτωση αγοράς αισθητήρα με έκδοση, 3 pin, δεν απαιτείται η χρήση αντίστασης καθώς είναι ενσωματωμένη στη πλακέτα.

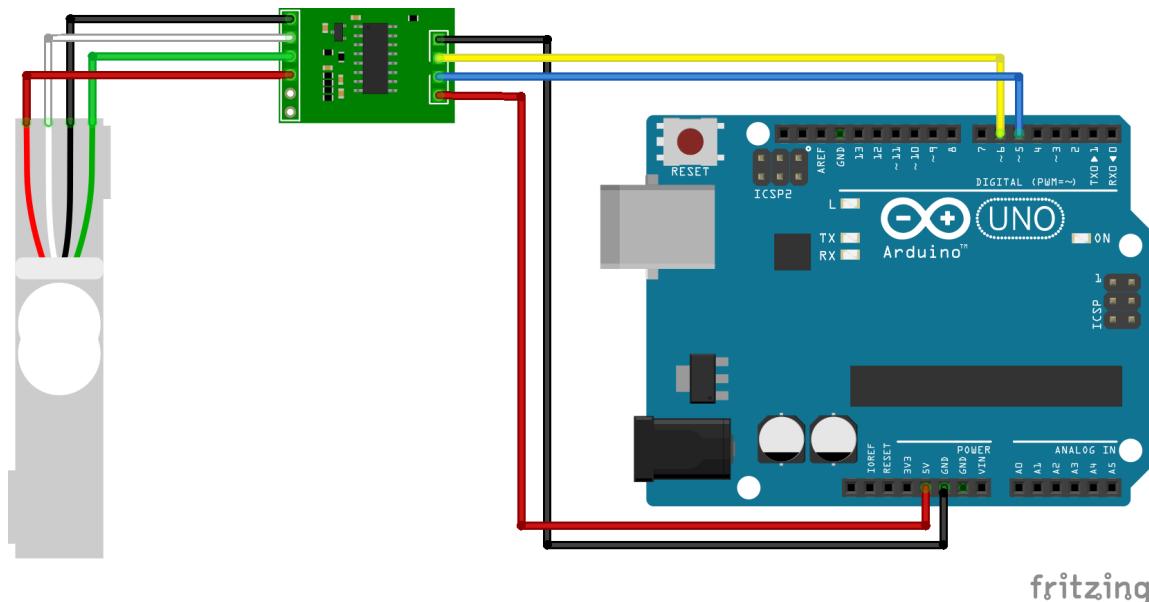


fritzing

Σχήμα 42: Συνδεσμολογία: DHT11 (3 pin) - Arduino

Κατά τη συνδεσμολογία του αισθητήρα χρειάζεται ιδιαίτερη προσοχή, διότι σε περίπτωση λανθασμένης σύνδεσης υπάρχει κίνδυνος καταστροφής του.

5.1.4.3 Συνδεσμολογία Αισθητήρα Βάρους - Μετατροπέα HX711



Σχήμα 43: Συνδεσμολογία: Αισθητήρα Βάρους - Μετατροπέα HX711 - Arduino

Κατά τη συνδεσμολογία του αισθητήρα χρειάζεται ιδιαίτερη προσοχή, διότι σε περίπτωση λανθασμένης σύνδεσης υπάρχει κίνδυνος καταστροφής του.

Σε περίπτωση μη λειτουργίας ή μηδενικής τιμής ή αρνητικής τιμής κατά την ανάγνωση τιμών από τον αισθητήρα, υπάρχει πρόβλημα στη συνδεσμολογία των καλωδίων.

Τα καλώδια του αισθητήρα βάρους είναι πολύ λεπτά, απαιτούνται λεπτοί χειρισμοί για τη πρόληψη ατυχημάτων.

Η συνδεσμολογία του αισθητήρα βάρους με το μετατροπέα HX711:

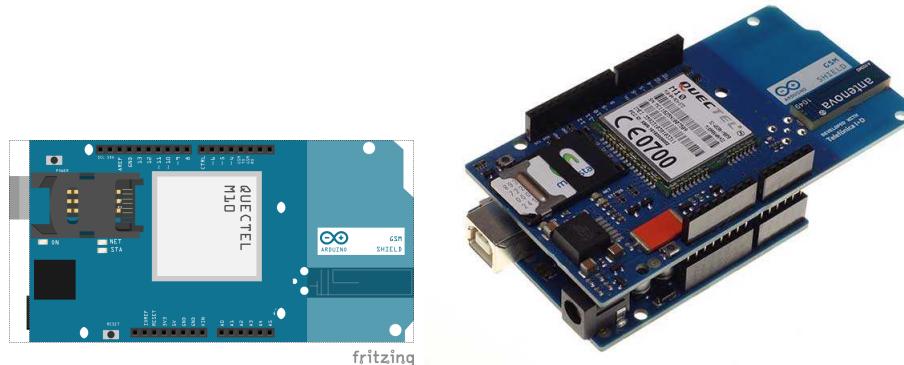
Από αισθητήρα βάρους:	Σε μετατροπέα HX711
E +	Μαύρο Καλώδιο
E -	Άσπρο Καλώδιο
A -	Κόκκινο Καλώδιο
A +	Πράσινο Καλώδιο

Η συνδεσμολογία του μετατροπέα HX711 με το Arduino:

Από μετατροπέα βάρους HX711:	Σε Arduino
GND (Ground)	Μαύρο Καλώδιο (Γείωση)
DT	Θέση 6 Arduino
SCK	Θέση 5 Arduino
VCC (5 Volt)	Κόκκινο Καλώδιο (Τροφοδοσία)

5.1.4.4 Συνδεσμολογία - GSM Shield

Η πλακέτα ²GSM Shield έχει εύκολη συνδεσμολογία, καθώς τα pins ταιριάζουν ακριβώς με τις υποδοχές του Arduino. Συνδέονται χωνευτά, επιτρέποντας έτσι την χρήση των υποδοχών, σα να υπήρχε μόνο η πλακέτα Arduino. Υπάρχει περιορισμός στις υποδοχές (pin 2 & 3) της χρησιμότητας τους για επικοινωνία μεταξύ της πλακέτας GSM και Arduino.

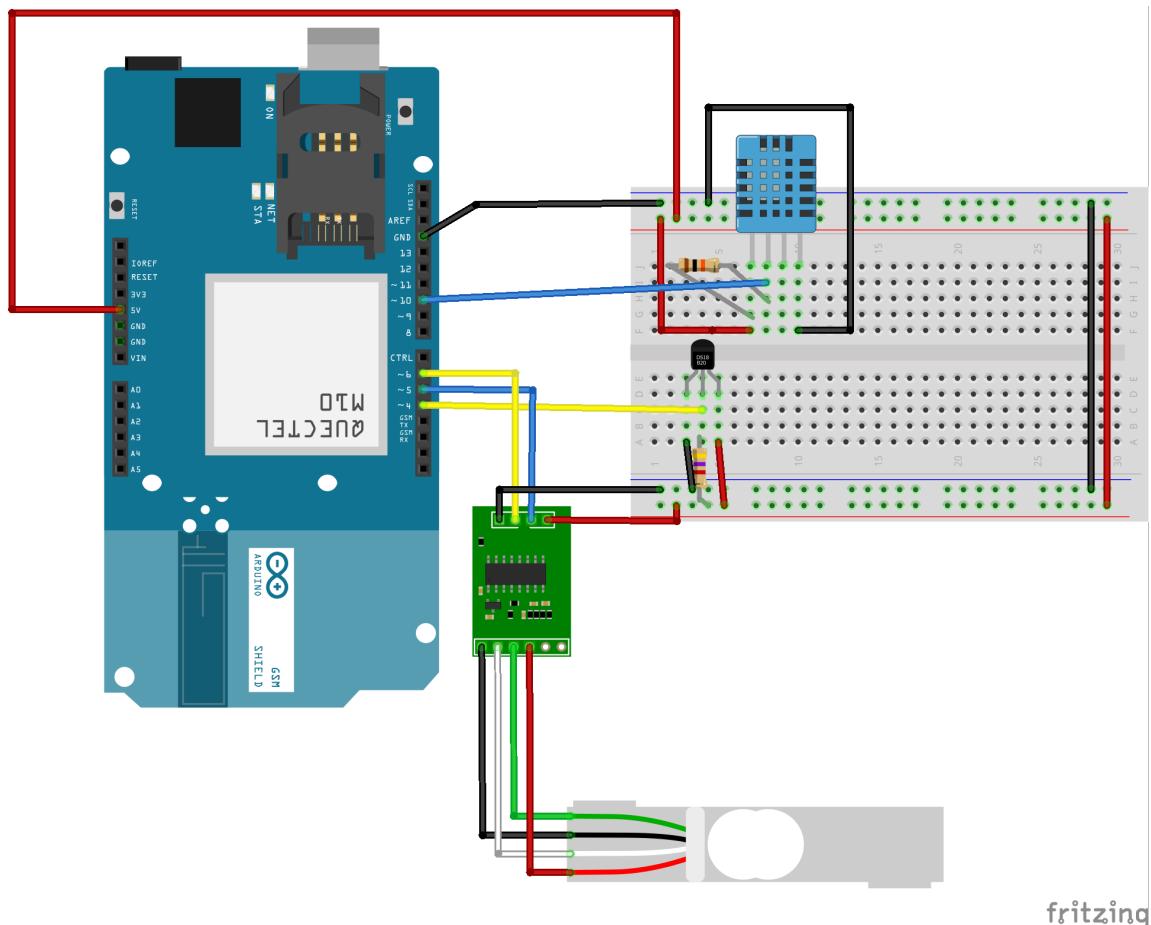


Σχήμα 44: Συνδεσμολογία: GSM Shield - Arduino

²Κατά τη σύνδεση της πλακέτας GSM δεν είναι διαθέσιμη η λειτουργία εγγραφής κώδικα στην πλακέτα του Arduino μέσω της θύρας USB.

5.1.4.5 Σχεδιάγραμμα Συνδέσεων Αισθητήρων

Ακολουθεί η τελική σύνδεση όλων των αισθητήρων.

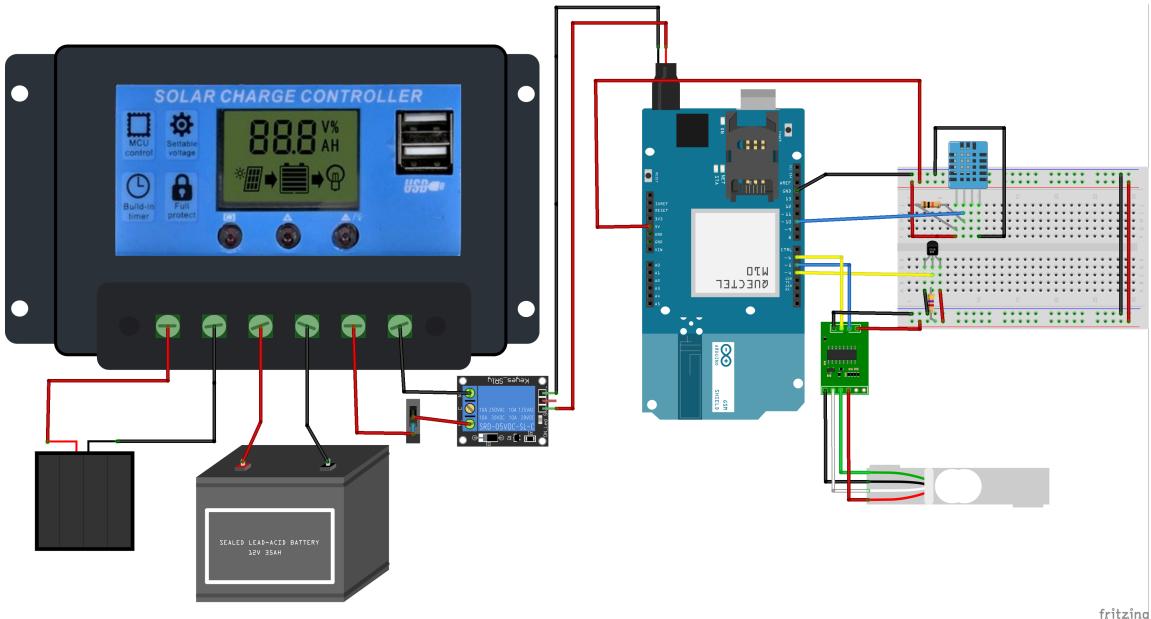


Σχήμα 45: Σχεδιάγραμμα Αισθητήρων (Τελική Μορφή)

Κατά τη συνδεσμολογία των αισθητήρων χρειάζεται ιδιαίτερη προσοχή, διότι σε περίπτωση λανθασμένης σύνδεσης υπάρχει κίνδυνος καταστροφής τους.

5.1.4.6 Τελική Συνδεσμολογία Όλων των Υλικών

Ακολουθεί η τελική σύνδεση όλων των προϊόντων για την υλοποιήση της ηλεκτρονικής ζυγαριάς.



Σχήμα 46: Σχεδιάγραμμα Κατασκευής (Τελική Μορφή)

Το ηλιακό πάνελ, συνδέεται στο ρυθμιστή φόρτισης (αριστέρη πλευρά υποδοχών). Η μπαταρία συνδέεται στη μεσαία πλευρά υποδοχών, η οποία περιβάλλεται από μια ασφάλεια των 20A, για τη φόρτιση της.

Στην έξοδο του ρυθμιστή συνδέεται ο χρονοδιακόπτης, (timer relay), ώστε να είναι η μόνη τροφοδοσία ρεύματος. Η τροφοδοσία για τη πλακέτα, παρεμβάλλεται από ένα διακόπτη για τον ασφαλή τερματισμό παροχής ενέργειας στο κύκλωμα.

Το κύκλωμα της ηλεκτρονικής ζυγαριάς, τροφοδοτείται με ρεύμα από την έξοδο του χρονοδιακόπτη, για καθορισμένη ώρα, η οποία έχει οριστεί από το προγραμματιστή. Για παράδειγμα, 10 λεπτά.

Έπειτα από καθορισμένο χρονικό διάστημα, το κύκλωμα παύει να έχει ρεύμα και έτσι τερματίζεται η λειτουργία του, δίνοντας μεγαλύτερη αυτονομία στο σύστημα. Με αυτό το τρόπο επιμηκύνεται η διάρκεια λειτουργίας του σε ημέρες με χαμηλή ηλιακή ακτινοβολία.

5.1.5 Κώδικας Δοκιμής

Ακολούθοι όλοι οι κώδικες για τις απαραίτητες δοκιμές των εξαρτημάτων και αισθητήρων.

5.1.5.1 Δοκιμή DS18B20

Για τη δοκιμή του αισθητήρα θερμοκρασίας ³:

- [TestDS18B20.ino](#)

```

1 #include <OneWire.h>
2 #include <DallasTemperature.h>
3
4 #define ONE_WIRE_BUS 4
5 OneWire oneWire(ONE_WIRE_BUS);
6 DallasTemperature sensors(&oneWire);
7
8 void setup(void)
9 {
10     Serial.begin(9600);
11     sensors.begin();
12 }
13 void loop(void)
14 {
15     sensors.requestTemperatures(); // Read temperature
16
17     Serial.print("Temperature is: ");
18     Serial.print(sensors.getTempCByIndex(0));
19     delay(1000);
20 }
```

Στο παραπάνω τμήμα κώδικα δοκιμάζετε η ομαλή λειτουργία του αισθητήρα θερμοκρασίας DS18B20. Στην αρχή πραγματοποιούνται, οι απαραίτητες δηλώσεις βιβλιοθηκών. Ορίζεται το pin, που θα διαβάζεται από τον αισθητήρα. Στη συνέχεια, η συνάρτηση setup() περιέχει την έναρξη λειτουργίας του αισθητήρα. Στη συνάρτηση loop() εκτελείται ο κώδικας, ο οποίος ζητάει δεδομένα από τον αισθητήρα, τη θερμοκρασία και

³Οι τιμές των θερμοκρασίων είναι σε κλίμακα Κελσίου. Σε περίπτωση επιθυμίας χρήσης βαθμών Φαρενάιτ αντί για Κελσίου, τότε στην γραμμή 20 αντικαθιστούμε την εντολή sensors.getTempCByIndex(0) με την εντολή sensors.getTempFByIndex(0) και η τιμή που λαμβάνεται από τον αισθητήρα θα είναι σε βαθμούς Φαρενάιτ.

την εμφανίζει στη ουδόνη του χρήστη. Η θερμοκρασία είναι με τη μορφή της κλίμακας Κελσίου. Με αυτό το πρόγραμμα ελέγχεται η ομαλή λειτουργία του αισθητήρα.

5.1.5.2 Δοκιμή DHT11:

Για τη δοκιμή του αισθητήρα υγρασίας και θερμοκρασίας⁴:

- [TestHumidity.ino](#)

```

1 #include <dht.h>
2
3 dht DHT;
4 #define DHT11_PIN 10
5
6 void setup(){
7     Serial.begin(9600);
8     delay(1000);
9 }
10
11 void loop(){
12     DHT.read11(DHT11_PIN);
13
14     Serial.print("Current humidity = ");
15     Serial.print(DHT.humidity);
16     Serial.print("% ");
17     Serial.print("temperature = ");
18     Serial.print(DHT.temperature);
19     Serial.println("C ");
20
21     delay(5000); //Wait 5 seconds.
22 }
```

Στο παραπάνω τμήμα κώδικα ελέγχεται η ομαλή λειτουργία του αισθητήρα υγρασίας και θερμοκρασίας DHT11. Στην αρχή του προγράμματος, γίνονται όλες οι απαραίτητες δηλώσεις βιβλιοθηκών και ο ορισμός της εισόδου δεδομένων από τον αισθητήρα. Στη συνέχεια, η συνάρτηση `setup()` περιέχει την έναρξη λειτουργίας του αισθητήρα.

Στη συνάρτηση `loop()` εκτελείται ο κώδικας ο οποίος ζητάει δεδομένα από τον αισθητήρα, για την υγρασία αλλά και τη θερμοκρασία. Τα δεδομένα αυτά, εμφανίζονται στην ουδόνη του χρήστη. Η θερμοκρασία είναι σε μορφή Κελσίου. Με αυτό το πρόγραμμα ελέγχεται η ομαλή λειτουργία του αισθητήρα.

⁴Οι τιμές των θερμοκρασίων είναι σε κλίμακα Κελσίου με απόκλιση ± 0.5 βαθμούς.

5.1.5.3 Δοκιμή HX711:

Για τη δοκιμή και τη ρύθμιση του αισθητήρα βάρους μέσω του μετατροπέα σήματος HX 711:

- [CalibrateWeightSensor.ino](#)

```

1 #include "HX711.h"
2
3 #define DOUT 5
4 #define CLK 6
5
6 HX711 scale;
7
8 float calibration_factor = 15550;
9
10 void setup() {
11     Serial.begin(9600);
12
13     scale.begin(DOUT, CLK);
14
15     long zero_factor = scale.read_average();
16     scale.set_scale(calibration_factor); //Adjust this value
17
18     int i;
19     for(i=0;i<100;i++)
20     {
21         Serial.print("Reading: ");
22         Serial.print(scale.get_units()/2.2046, 4);
23         Serial.print(" kg");
24         Serial.print(" calibration_factor: ");
25         Serial.println(calibration_factor);
26     }
27     Serial.println();
28     Serial.print("Reading: ");
29     Serial.print(scale.get_units()/2.2046, 4);
30     Serial.print(" kg");
31     Serial.print(" calibration_factor: ");
32     Serial.println(calibration_factor);
33 }
```

```

34
35 void loop() {
36 }
```

Στην αρχή του προγράμματος γίνεται εισαγωγή της βιβλιοθήκης με όλες τις απαραίτητες συναρτήσεις και δηλώσεις. Στις γραμμές, 3 και 4, γίνεται δήλωση `pin` που χρησιμοποιεί ο αισθητήρας HX711.

Στη συνέχεια, δηλώνονται μεταβλητές για τη πρόσβαση στα δεδομένα και μια μεταβλητή. Η μεταβλητή αυτή αποθηκεύει τον αριθμό (calibration factor) που ρυθμίζει τον αισθητήρα βάρους να διαβάζει σωστά.

Στη συνάρτηση `setup()`, γίνονται οι κατάλληλες αρχικοποιήσεις, που δηλώνονται οι θέσεις σύνδεσης του μετατροπέα. Ορίζεται στο μετατροπέα ο αριθμός ρύθμισης για τη σωστή μέτρηση του βάρους. (Σε αυτό το σημείο πραγματοποιούνται οι αλλαγές με αυτόν τον αριθμό).

Για την εύρεση του κατάλληλου αριθμού εκτελείται κώδικας ο οποίος δείχνει στην οθόνη του χρήστη, μέσω του Arduino Ide, τη μέτρηση βάρους του αντικειμένου που έχει τοποθετηθεί στην ζυγαριά. (Κατασκευή από σίδερα, που περιλαμβάνει τον αισθητήρα βάρους και το μετατροπέα).

Σε αυτό το σημείο, ο κώδικας εκτελείται αρκετές φορές μέχρι να βρεθεί το βάρος, το οποίο είναι ήδη γνωστό. Στις διάφορες μετρήσεις που εκτελέστηκαν, ο κατάλληλος αριθμός, βρέθηκε στη τιμή, 15550. Αυτός ο αριθμός αλλάζει από αισθητήρα σε αισθητήρα και απαιτείται η εκτέλεση του παραπάνω τμήματος κώδικα.

5.1.5.4 Δοκιμή GSM Shield

Για τον έλεγχο και τη ρύθμιση αποστολής γραπτού μηνύματος SMS :

- [TestGSMShield.ino](#)

```

1 include <GSM.h>
2
3 #define PINNUMBER ""
4 // If your SIM has pin
5 // change this line from ""
6 // to your PIN NUMBER
7 // e.g: #define PINNUMBER "1234"
8
9 GSM gsmAccess;
10 GSM_SMS sms;
11 char remoteNumber[20]= "+3069XXXXXXX";
12 char txtMsg[200]="Test ";
13
14 void setup()
15 {
16     Serial.begin(9600);
17     Serial.println("SMS Messages Sender");
18     boolean notConnected = true;
19     // If your SIM has PIN,
20     //pass it as a parameter of begin() in quotes
21
22     while(notConnected)
23     {
24         if(gsmAccess.begin(PINNUMBER)==GSM_READY)
25             notConnected = false;
26         else
27         {
28             Serial.println("Not connected");
29             delay(1000);
30         }
31     }
32     Serial.println("GSM initialized");
33     sendSMS();
34 }
```

```

36 void loop()
37 {
38 // nothing to see here
39 }
40
41 void sendSMS() {
42   Serial.println("Message: ");
43   Serial.println(txtMsg);
44   sms.beginSMS(remoteNumber);
45   sms.print(txtMsg);
46   sms.endSMS();
47   Serial.println("\nCOMPLETE!\n");
48 }
```

Στην αρχή του προγράμματος, γίνεται δήλωση των απαραίτητων βιβλιοθηκών και μεταβλητών. Κάποιες από αυτές είναι, ο κωδικός της κάρτας SIM, ακόμη και αν δεν υπάρχει (δηλώνεται κενή μεταβλητή), ο αριθμός κινητού του παραλήπτη και το κείμενο που θα περιέχει το γραπτό μήνυμα.

Στη συνάρτηση `setup()`, γίνονται οι κατάλληλες ενέργειες για τη σύνδεση της πλακέτας GSM μέχρι να γίνει η σύνδεση της στο δίκτυο κινητής τηλεφωνίας. Σε περίπτωση αποτυχημένης προσπάθειας εισόδου κάρτας SIM γίνεται προσπάθεια επαναλλημένα από το πρόγραμμα μέχρι να καταφέρει να γίνει η σύνδεση στο δίκτυο.

Αυτή η ενέργεια, μπορεί να προκαλέσει το 'κλείδωμα της κάρτας', όπως θα συνέβαινε αν ένας χρήστης πληκτρολογούσε τον κωδικό της κάρτας SIM στη κινητή του συσκευή, τρεις φόρες λανθασμένα.

Εφόσον έχει γίνει η σύνδεση με το δίκτυο της κινητής τηλεφωνίας, το πρόγραμμα προχωράει με τη συνάρτηση `sendSMS()`, η οποία στέλνει γραπτό μήνυμα στον ορισμένο τηλεφωνικό αριθμό. Όταν ολοκληρωθεί η αποστολή του SMS, γίνεται ενημέρωση του αποστολέα ότι το πρόγραμμα ολοκληρώθηκε.

Σε αυτό το σημείο, έχει επέλθει το πέρας του προγράμματος, καθώς η συνάρτηση `loop()` δεν περιέχει κάποιο τμήμα κώδικα για να εκτελεστεί. Ο έλεγχος της συσκευής GSM ολοκληρώνεται με την επαλήθευση της αποστολής γραπτού μηνύματος στο δηλωθέν παραλήπτη.

5.1.6 Τελική Μορφή Κώδικα

Το τελικό πρόγραμμα το οποίο αναπτύχθηκε:

- [RelayBeeHiveSystem.ino](#)

```

1 //=====
2 //=
3 //=====
4 /*Include Libraries*/
5 #include <stdio.h>
6 #include <stdlib.h>
7 #include <string.h>
8 #include <GSM.h>
9 #include <OneWire.h>
10 #include <DallasTemperature.h>
11 #include <dht.h>
12 #include "HX711.h"

```

Το αρχείο ξεκινάει με δηλώσεις των βιβλιοθηκών.

```

13 /*Define pins*/
14 //DS1820B
15 #define ONE_WIRE_BUS 4 //kitrino
16 OneWire oneWire(ONE_WIRE_BUS);
17 DallasTemperature sensors(&oneWire);
18
19 //GSM SIM CARD
20 #define PINNUMBER ""
21
22 //Varos
23 #define DOUT 5 //mple
24 #define CLK 6 //kitrino
25
26 //DTH11
27 dht DHT;
28 #define DHT11_PIN 10 //mple

```

Στη συνέχεια, δηλώνονται οι μεταβλητές που αντιστοιχούν στις εξόδους των αισθητήρων, ώστε να γνωρίζει το **Arduino** από που θα προέρχονται τα δεδομένα. Σε αυτό το σημείο, δηλώνονται τα pins του αισθητήρα της θερμοκρασίας (**DS1820B**), του βάρους και τέλος του αισθητήρα της υγρασίας (**DHT11**).

Ακόμη δηλώνεται ο κωδικός πρόσβασης (PIN) της κάρτα SIM.
Για την αποφυγή τυχόν κλειδώματος της κάρτας, από τη πλακέτα επικοινωνίας μπορεί να είναι κενός ο κωδικός αυτός (δεν προτείνεται).

```

29 //=
30 //=                               GLOBALS = =
31 //=
32 HX711 scale;
33 float calibration_factor = 15550; //old = 12880
34
35 /* Globals*/
36 GSM gsm;
37 GSM_SMS sms;
38 float temperature1; /*DS1820B*/
39 float temperature2; /*DTH11*/
40 float temperature;
41 float weight;
42 float metrisi;
43 int humidity;
44 int readData; /*Humidity check data*/
45 int flag_for_v = 0; /*flag for metrisi varous*/
46 char phone[20] = "+3069XXXXXXX";
47 char msg[160]; /*Message*/
48 char str[30]; /*helpful array for SMS*/
49 int i=0; /*counter*/
50 int j=0; /*counter for sms*/

```

Δηλώνονται οι κατάλληλες μεταβλητές που απαιτούνται. Μέρος αυτών αρχικοποιούνται κατά την δήλωση τους, όπως για παράδειγμα, ο παράγοντας ‘καλιμπραρίσματος’ (calibration factor).

Ακόμη, δηλώνεται το τηλέφωνο, όπου θα γίνει η αποστολή του SMS σε ένα πίνακα χαρακτήρων 20 θέσεων. Είναι απαραίτητο να δηλωθεί ο κωδικός της χώρας. Για την Ελλάδα, ο κωδικός χώρας είναι: +30.

Μορφή SMS:

Thermokrasia: 22.13 C
Ygrasia: 43 %
Weight: 34.03 kg

Στη συνέχεια του προγράμματος, εκτελούνται οι δύο βασικές συναρτήσεις **setup()**, η οποία εκτελείται μία φορά κατά την αρχή και η **loop()**, η οποία εκτελείται συνέχεια. Στη συγκεκριμένη περίπτωση δεν υπάρχει τμήμα κώδικα που να επαναλαμβάνεται καθώς δεν το απαιτούν οι ανάγκες του συστήματος.

```

51 //=====
52 //= PROGRAM =
53 //=====
54 void setup()
55 {
56     arxi();
57     delay(10000);
58
59     /*Power up sensors*/
60     scale.begin(DOUT, CLK);
61     long zero_factor = scale.read_average();
62     scale.set_scale(calibration_factor);
63
64     readData = DHT.read11(DHT11_PIN);
65
66     sensors.begin();
67     metrao_ugrasia();
68     metrao_ther();
69     metrao_varos();
70     stelno_sms();
71 }
72
73 void loop()
74 {
75 }
76 }
```

Η συνάρτηση **setup()** είναι απλοποιημένη για τη καλύτερη δυνατή ανάγνωση και οργάνωση του κώδικα. Χωρίζεται σε διάφορες συναρτήσεις με ονόματα που ο χρήστης κατανοεί άμεσα τη λειτουργία τους.

Η συνάρτηση **arxi()** δηλώνει και την έναρξη εκτέλεσης του προγράμματος.

```

76 void arxi()
77 {
78     boolean notConnected = true;
79
80     Serial.begin(9600);
81     while (!Serial)
82     {
83         ; /*waiting serial to connect*/
84     }
85     Serial.println("Bee Hive System");
86     /*Start GSM*/
87
88     while(notConnected)
89     {
90         if(gsm.begin(PINNUMBER) == GSM_READY)
91         {
92             notConnected = false;
93         }
94         else
95         {
96             Serial.println("GSM Not Connected");
97             delay(1000);
98         }
99     }
100
101    /*GSM Shied is connected*/
102    Serial.println("GSM initialized");
103 }
```

Αφορά τη σύνδεση της GSM πλακέτας με το δίκτυο 2G/3G αναλόγως την υποστήριξη της περιοχής που βρίσκεται τοποθετημένη η ηλεκτρονική ζυγαριά. Η ηλεκτρονική ζυγαριά εκκινεί τη πλακέτα GSM, δηλώνοντας της το κωδικό PIN. Η προσπάθεια γίνεται μέχρι ότους η συσκευή κατάφερε να εκκινήσει επιτυχώς. Σε περίπτωση που ο κωδικός PIN είναι εσφαλμένος η κάρτα SIM κλειδώνει και απαιτείται η σύνδεση της με κινητή συσκευή για εισαγωγή κωδικού PUK.

Στη συνέχεια το πρόγραμμα προχωράει με αρχικοποίηση του αισθητήρα του βάρους, όπου ορίζεται η μεταβλητή του καλιμπραφίσματος για να γίνει η σωστή μέτρηση βάρους. (Γραμμές Κώδικα: 59-62)

```

104 void metrao_ugrasia()
105 {
106     humidity = DHT.humidity;
107     Serial.print("Humidity : ");
108     Serial.println(humidity);
109 }
```

Γίνεται η αρχικοποίηση των αισθητηρών, η μέτρηση τιμών ξεχινά από τον αισθητήρα υγρασίας, **DHT11**, η συνάρτηση **metrao_ugrasia()** διαβάζει, τη τιμή της υγρασίας που υπάρχει στο χώρο και αποθηκεύεται στη μεταβλητή **humidity** ώστε να χρησιμοποιηθεί αργότερα.

```

110 void metrao_ther()
111 {
112     sensors.requestTemperatures();
113     temperature1 = sensors.getTempCByIndex(0);
114     Serial.print("Temperature 1 : ");
115     Serial.println(temperature1);
116     temperature2 = DHT.temperature;
117     Serial.print("Temperature 2 : ");
118     Serial.println(temperature2);
119
120     /*Finalize Temperature*/
121     if(temperature2 > temperature1)
122     {
123         temperature = temperature1;
124     }
125     else
126     {
127         if(temperature1 < 0){
128             temperature = temperature2;
129         }
130         else{
131             temperature = temperature1;
132         }
133     }
134 }
```

Το επόμενο βήμα μετά τη μέτρηση της υγρασίας, είναι η μέτρηση της θερμοκρασίας. Στην ηλεκτρονική ζυγαριά υπάρχουν δύο αισθητήρες θερμοκρασίας. Ένας αποκλειστικός αισθητήρας θερμοκρασίας, ο **DS18B20** και ο δεύτερος αισθητήρας θερμοκρασίας,

DHT11. Σε αυτή τη περίπτωση έχουμε δύο μετρήσεις θερμοκρασίας με ελάχιστη απόκλιση, λόγω διαφοράς κατασκευής τους.

Σε περίπτωση απώλειας του ενός αισθητήρα, υπάρχει η δυνατότητα ακόμη εύρεσης της τιμής της θερμοκρασίας του περιβάλλοντος.

Η μέτρηση αυτή μαζί σε συνδυασμό με τον αισθητήρα της υγρασίας, προσδίνουν χρήσιμα δεδομένα για τις μεταβολές των περιβαλλοντικών συνθηκών του μικροκλίματος της περιοχής, στο μελισσοκόμο.

Αυτό σηματοδοτεί συνήθως την απότομη ελάττωση παραγωγής, με αποτέλεσμα η κυψέλη να σταματήσει τη παραγωγή μελιού. Σε αυτό το σημείο ζεκινάει η κατανάλωση του μελιού προκαλώντας ζημία στη παραγωγή.

```

135 void metrao_varos()
136 {
137     int i = 0;
138     for (i=0;i<200;i++)
139     {
140         weight = scale.get_units() / 2.2046;
141     }
142     weight = scale.get_units() / 2.2046;
143     weight = scale.get_units() / 2.2046;
144     Serial.print("Metrao Varos ");
145     Serial.print(weight, 3);
146     Serial.println(" kgs");
147 }
```

Στη συνέχεια, γίνεται η μέτρηση του βάρους μέσω της συνάρτησης **metrao_varos()**. Γίνεται κατ' επανάληψη, 200 φορών για την εξάλειψη τυχόν θορύβων και την εξασφάλιση σωστής μέτρησης.

Απαιτείται η χρήση, μεθόδου (formulas), για την μετατροπή από lbs σε kg και ο τύπος είναι: $kg = lbs \div 2.2046$

```

148 void stelno_sms()
149 {
150     strcpy(str, "Thermokrasia: ");
151     j=0;
152     for(;j<strlen(str);i++)
153     {
154         msg[i] = str[j];
155         j++;
156     }
157     dtostrf(temperature, 4, 2, str);
158     j=0;
159     for(;j<strlen(str);i++)
160     {
161         msg[i] = str[j];
162         j++;
163     }
164     strcpy(str, " C\nYgrasia: ");
165     j=0;
166     for(;j<strlen(str);i++)
167     {
168         msg[i] = str[j];
169         j++;
170     }
171     sprintf(str, "%d %%", humidity);
172     j=0;
173     for(;j<strlen(str);i++)
174     {
175         msg[i] = str[j];
176         j++;
177     }
178     strcpy(str, "\nWeight: ");
179     j=0;
180     for(;j<strlen(str);i++)
181     {
182         msg[i] = str[j];
183         j++;
184     }

```

```

185     dtostrf( weight , 4, 2, str );
186     j=0;
187     for ( ;j<strlen ( str ) ; i++)
188     {
189         msg [ i ] = str [ j ];
190         j++;
191     }
192     strcpy ( str , " kg" );
193     j=0;
194     for ( ;j<strlen ( str ) ; i++)
195     {
196         msg [ i ] = str [ j ];
197         j++;
198     }
199     for ( j=i ;j<strlen ( msg ) ; j++)
200     {
201         msg [ i ] = '\0';
202     }
203     delay(2000);
204 //   Serial . println (msg);
205     int sms_flag = 1;
206     /*Send Message*/
207     sms . beginSMS ( phone );
208     sms . print ( msg );
209     while ( true ){
210         sms_flag = sms . endSMS ( );
211         Serial . print ( "Sms Flag is : " );
212         Serial . println ( sms_flag );
213         if ( sms_flag == 1){
214             break;
215         }
216     }
217     Serial . println ( "SMS ——————" );
218     Serial . println ( msg );
219     Serial . println ( "End ——————" );
220 }
```

Τέλος, εφόσον υπάρχουν όλες οι μετρήσεις, συντάσσεται το μήνυμα SMS μέσω της συνάρτησης **stelno_sms()**. Χρησιμοποιείται μια μεταβλητή τύπου αλφαριθμητικού (str) για την υποβοήθηση της αποθήκευση ολόκληρου του SMS. Το μήνυμα ξεκινάει με

τη μέτρηση της θερμοκρασίας. Η μεταβλητή `j` αρχικοποιείται με τη τιμή 0. Ακολουθεί μια επανάληψη από το 0 έως το συνολικό μήκος του αλφαριθμητικού (str). Γίνεται αντιγραφή του προσωρινού κειμένου (str) στη μεταβλητή που κρατάει το συνολικό κείμενο (msg). Στο τελικό μήνυμα, συμπεριλαμβάνονται οι μετρήσεις της θερμοκρασίας, της υγρασίας και τέλος του βάρους.

Γίνεται η σύνταξη του κειμένου και άμεσα ξεκινάει η αποστολή στο κινητό τηλέφωνο που έχει οριστεί. Η συνάρτηση αποστολής βρίσκεται σε συνεχόμενη επανάληψη μέχρι η ένδειξη αποστολής γίνει αληθής στον έλεγχο. Με αυτό το τρόπο, εξασφαλίζεται η αποστολή του μηνύματος μια φορά, με σκοπό τη αποφυγή της μη αποστολής ή επαναλαμβανόμενων αποστολών προκαλώντας έξτρα χρεώσεις.

Μόλις τελειώσει και η τελευταία συνάρτηση τότε το πρόγραμμα εισέρχεται στη συνάρτηση `loop()` πραγματοποιώντας ατέρμονα επαναλήψεις, μέχρι να επέλθει η διακοπή παροχής ενέργειας από το χρονοδιακόπτη.

5.2 Website

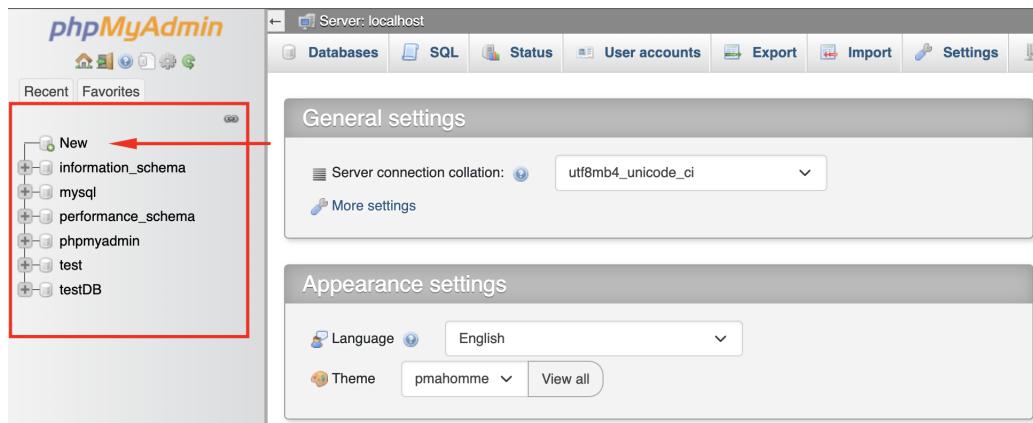
Για τις ανάγκες ανάπτυξης της ιστοσελίδας, χρησιμοποιήθηκε τοπικός server που παρέχεται από το πρόγραμμα, XAMPP.

5.2.1 Δημιουργία Βάσης Δεδομένων

Η ανάγκη αποθήκευσης δεδομένων, απαιτεί τη δημιουργία βάσης δεδομένων (**MySQL**). Η βάση μπορεί να δημιουργηθεί είτε μέσω του διαχειριστικού περιβάλλοντος (**GUI**), είτε μέσω γραμμή εντολών (**terminal**).

Προτείνεται η δημιουργία να γίνει μέσα από το διαχειριστικό περιβάλλον των βάσεων δεδομένων του Xampp, στην ιστοσελίδα, <http://localhost/phpmyadmin/>.

1. Από το μενού στο πλάι, επιλέγετε το κουμπί **New** (New) για να ξεκινήσει η δημιουργία νέας βάσης δεδομένων.



Σχήμα 47: Δημιουργία Βάσης Δεδομένων

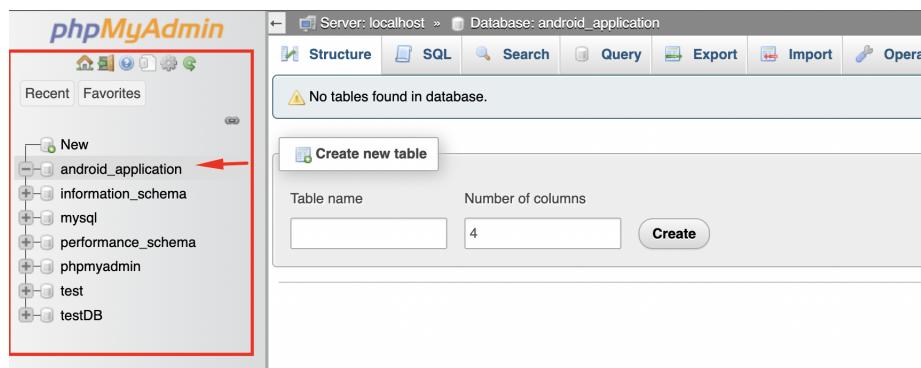
2. Στην επόμενη σελίδα, συμπληρώνεται το όνομα και το τύπο δεδομένων που θα αποθηκευτούν. Για τη χρήση της ελληνικής γλώσσας, πρέπει ο τύπος της βάσης να είναι μορφής, **utf8mb4_general_ci**. Το όνομα της βάσης, πρέπει να είναι αντιπροσωπευτικό σε σχεσή με το θέμα.

Databases



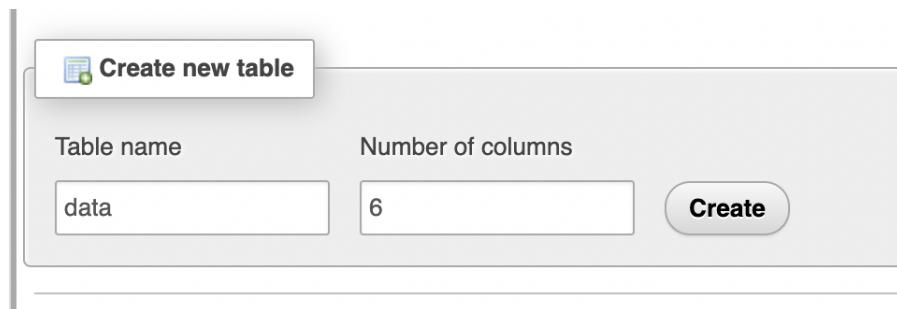
Σχήμα 48: Όνομα και Τύπος Βάσης Δεδομένων

- Στην επόμενη σελίδα, η βάση με όνομα, **android_application** έχει δημιουργηθεί. Το επόμενο βήμα είναι η δημιουργία πίνακα. Σε αυτό το σημείο, ο χρήστης θα πρέπει να σκεφτεί με ποιον τρόπο θα αποθηκεύονται τα δεδομένα.



Σχήμα 49: Δημιουργία Νέας Βάσης Δεδομένων

- Στο πρώτο πίνακα, θα χρειαστεί να αποθηκευτούν τα δεδομένα που στέλνει η ηλεκτρονική ζυγαριά. Η ονομασία του, **data**. Ο πίνακας, θα περιέχει ένα αναγνωριστικό πεδίο, το οποίο θα είναι και το πρωτεύον κλειδί και πέντε ακόμη πεδία που θα αποθηκεύει το όνομα του χρήστη, την ημερομηνία, τη ωραϊκότητα, την υγρασία και τέλος το βάρος.



Σχήμα 50: Επιλογή ονόματος και αριθμός στηλών πίνακα data

5. Για την δομή του πίνακα:

Όνομα Τύπου	Τύπος	Μέγεθος	Πρωτεύον	Κλειδί
id	INT	11	Ναι	
user	VARCHAR ⁵	50	-	
imerominia	DATE	-	-	
thermokrasia	FLOAT	-	-	
ugrasia	FLOAT	-	-	
baros	FLOAT	-	-	

⁵Όταν πρόκεται για μεταβλητές που αποθηκεύουν αλφαριθμητικά, μια καλή τεχνική για την χρήση λιγότερου χώρου στη βάση δεδομένων είναι να χρησιμοποιείται ο τύπος, VARCHAR, ο οποίος δεσμεύει όσα στοιχεία χαρακτήρων, χρειάζεται μέχρι το μέγιστο μέγεθος που έχει οριστεί.

Σε αυτή τη περίπτωση, αν ο χρήστης έχει όνομα με μέγεθος χαρακτήρων 30, το μέγιστο όριο χαρακτήρων που μπορεί να χρησιμοποιήσει είναι εως και 50 χαρακτήρες.

Στη βάση δεδομένων, θα αποθηκευτεί το όνομα με μέγεθος 30 χαρακτήρων και όχι με το μέγιστο όριο, δηλαδή των 50 χαρακτήρων. Με αυτό τον τρόπο επιτυγχάνεται η εξασφάλιση μείωσης χώρου.

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I
id	INT	11	None			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/> PRIMARY
user	VARCHAR	50	None			<input type="checkbox"/>	---	<input type="checkbox"/>
imerominia	DATE		None			<input type="checkbox"/>	---	<input type="checkbox"/>
thermokrasia	FLOAT		None			<input type="checkbox"/>	---	<input type="checkbox"/>
ugrasia	FLOAT		None			<input type="checkbox"/>	---	<input type="checkbox"/>
baros	FLOAT		None			<input type="checkbox"/>	---	<input type="checkbox"/>

Σχήμα 51: Επιλογή δομής πίνακα data

Σε περίπτωση χρήσης εντολών, αντί του γραφικού περιβάλλοντος, ο κώδικας δημιουργίας του πίνακα μέσω γραμμής εντολών:

```
CREATE TABLE `android_application`.`data`
(`id` INT NOT NULL AUTO_INCREMENT ,
`user` VARCHAR(50) NOT NULL ,
`imerominia` DATE NOT NULL ,
`thermokrasia` FLOAT NOT NULL ,
`ugrasia` FLOAT NOT NULL ,
`baros` FLOAT NOT NULL ,
PRIMARY KEY (`id`)) ENGINE = InnoDB;
```

6. Ακόμη, πρέπει να δημιουργηθεί άλλος ένας πίνακας που θα διαχειρίζεται τους χρήστες της εφαρμογής. Η ονομασία του θα είναι **users**, επειδή αφορά τους χρήστες. Ένα πεδίο θα είναι ένας αύξων αριθμός που θα αποτελεί και το πρωτεύον κλειδί και τα υπόλοιπα αφορούν το όνομα, το επώνυμο, το username (όνομα χρήστη), το κωδικό πρόσβασης του χρήστη, το τηλέφωνο και τέλος το email του.

7. Για την δομή του πίνακα:

Όνομα Τύπου	Τύπος	Μέγεθος	Πρωτεύον Κλειδί
id	INT	15	Ναι
username	VARCHAR	50	-
password	VARCHAR	255	-
phone	TEXT	-	-
gui	TINYINT	1	-
name	VARCHAR	20	-
surname	VARCHAR	20	-
email	VARCHAR	50	-

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.
id	INT	15	None			PRIMARY		<input checked="" type="checkbox"/>
username	VARCHAR	50	None					
password	VARCHAR	255	None					
phone	TEXT		None					
gui	TINYINT	1	None					
name	VARCHAR	20	None					
surname	VARCHAR	20	None					
email	VARCHAR	50	None					

Σχήμα 52: Επιλογή δομής πίνακα users

Σε περίπτωση χρήσης εντολών αντί γραφικού περιβάλλοντος, ο κώδικας δημιουργίας του πίνακα μέσω γραμμής εντολών:

```
CREATE TABLE `android_application`.`users`
(`id` INT(15) NOT NULL AUTO_INCREMENT ,
`username` VARCHAR(50) NOT NULL ,
`password` VARCHAR(255) NOT NULL ,
```

```

`phone` TEXT NOT NULL ,
`gui` TINYINT(1) NOT NULL ,
`name` VARCHAR(20) NOT NULL ,
`surname` VARCHAR(20) NOT NULL ,
`email` VARCHAR(50) NOT NULL ,
PRIMARY KEY (`id`) ENGINE = InnoDB;

```

5.2.2 Πρόγραμμα Παραγωγής τυχαίων δεδομένων

Σε αυτό το σημείο έχει δημιουργηθεί μια βάση δεδομένων, η οποία περιέχει, δύο πίνακες, ένας για δεδομένα και ένας για τους χρήστες. Αναπτύχθηκε πρόγραμμα (σε Python) που παράγει τυχαία δεδομένα για τη συμπλήρωση του πίνακα, **data** για τις ανάγκες παρουσίασης.

- [FillDataToCSV.py](#)

```

1 #                                                 #
2 # Fill up demo data - Program                 #
3 #                                                 #
4
5 import random
6 import datetime
7 import csv
8
9 def exportCSV(row):
10    with open('data.csv', 'w', newline='') as file:
11        writer = csv.writer(file)
12        writer.writerow(row)
13
14 # Create random date
15 def getMonth(month, final_days):
16    values = list()
17    for i in range(1,final_days):
18        values.append([0,"sof","2022" + "/" + str(month) + "/"
19                      + str(i),getTemp(),getHum(),getWei()])
20
21 # Get random number representing temperature
22 def getTemp():

```

```

23     return round(random.uniform(1,48),2)
24
25 # Get random number representing humidity
26 def getHum():
27     return round(random.uniform(1,100),2)
28
29 # Get random number representing weight
30 def getWei():
31     return round(random.uniform(1,170),2)
32
33 def start():
34     data = list()
35     for i in range(1,13):
36         #Month 31
37         if i == 1 or i == 3 or i == 5 or i == 7 or i == 8 or i
38             == 10 or i == 12:
39             data = data + getMonth(i,32)
40         #Month 30
41         elif i == 4 or i == 6 or i == 9 or i == 11:
42             data = data + getMonth(i,31)
43         # February 28 days
44         elif i == 2:
45             data = data + getMonth(i,29)
46     exportCSV(data)
47 if __name__ == "__main__":
48     start()

```

Το πρόγραμμα παράγει ένα αρχείο μορφής .csv⁶ για να γίνει εισαγωγή δεδομένων από αυτό στη βάση δεδομένων. Στην αρχή του προγράμματος, δηλώνονται οι απαραίτητες βιβλιοθήκες. Στη συνέχεια, δηλώνονται και υλοποιούνται οι συναρτήσεις που αποσκοπούν σε κάποια συγκεκριμένη λειτουργία, σύμφωνα με το όνομα τους.

Η συνάρτηση **exportCSV(row)**, δημιουργεί ένα αρχείο με όνομα **data.csv** και εκεί εγγράφει τη λίστα (row) που δέχτηκε σαν παράμετρο. Η συνάρτηση **getMonth(month, final_days)** δέχεται δύο παραμέτρους, η μια αντιπροσωπεύει το μήνα και η δεύτερη τις ημέρες που αντιστοιχούν σε αυτόν.

Δημιουργεί μια λίστα με τιμές οι οποίες περιέχουν τις ημέρες του μήνα, ξεκινώντας από τη πρώτη μέρα έως και τη τελευταία ημέρα του μήνα (**final_days**), θερμοκρασία (**getTemp()**), υγρασία (**getHum()**) και βάρος **getWei()**.

Η συνάρτηση **getTemp()**, επιστρέφει μια τυχαία τιμή ως θερμοκρασία, μεταξύ των

τιμών, 1 έως 48, με ακρίβεια δύο δεκαδικών ψηφίων. Η συνάρτηση `getHum()`, επιστρέφει μια τυχαία τιμή ως υγρασία, μεταξύ των τιμών, 1 έως 100. Η συνάρτηση `(getWei())` επιστρέφει μια τυχαία τιμή ως βάρος, μεταξύ των τιμών, 1 εως 170. Οι τιμές αυτές προσομοιώνουν τις τιμές που υπάρχουν στο περιβάλλον αλλά και στη κυψέλη.

	A
1	0,sof,2022/1/1,1,2,22,9,39.57
2	0,sof,2022/1/2,23,26,12,4,158.04
3	0,sof,2022/1/3,28,53,59,27,65.22
4	0,sof,2022/1/4,18,41,71,94,45.68
5	0,sof,2022/1/5,11,57,67,94,21.73
6	0,sof,2022/1/6,43,21,43,4,15.11
7	0,sof,2022/1/7,45,71,2,19,112.33
8	0,sof,2022/1/8,10,56,29,66,138.63
9	0,sof,2022/1/9,24,03,30,19,151.98
10	0,sof,2022/1/10,18,47,69,38,77.19
11	0,sof,2022/1/11,34,73,22,61,87.42
12	0,sof,2022/1/12,47,86,49,68,152.35
13	0,sof,2022/1/13,34,93,97,21,161.52
14	0,sof,2022/1/14,40,81,75,36,158.8
15	0,sof,2022/1/15,18,61,52,61,96.01
16	0,sof,2022/1/16,32,41,49,83,164.64
17	0,sof,2022/1/17,36,18,18,2,112.44
18	0,sof,2022/1/18,43,96,76,26,38.82
19	0,sof,2022/1/19,38,55,76,98,164.75
20	0,sof,2022/1/20,37,93,90,35,131.56
21	0,sof,2022/1/21,33,05,90,3,35.24
22	0,sof,2022/1/22,21,27,22,73,45.3
23	0,sof,2022/1/23,29,36,16,95,66.23

Σχήμα 53: Μορφή αρχείου .csv

Το αρχείο που δημιουργήθηκε είναι έτοιμο να γίνει εισαγωγή στη βάση δεδομένων. Από τη σελίδα διαχείρισης της βάσης, επιλέγεται αρχικά ο πίνακας **data**. Από τις επιλογές στο πάνω μέρος, καρτέλα **Εισαγωγή (Import)**. Επιλέγεται η μορφή αρχείου, **csv**, από την αντίστοιχη ενότητα.

Εντοπίζεται το αρχείο και ξεκινάει η διαδικασία μέσω της **Εισαγωγής (Import)**. Σε αυτό το σημείο έγιναν εισαγωγή 365 εγγραφές, με τυχαία δεδομένα για κάθε ημέρα για το έτος 2022.

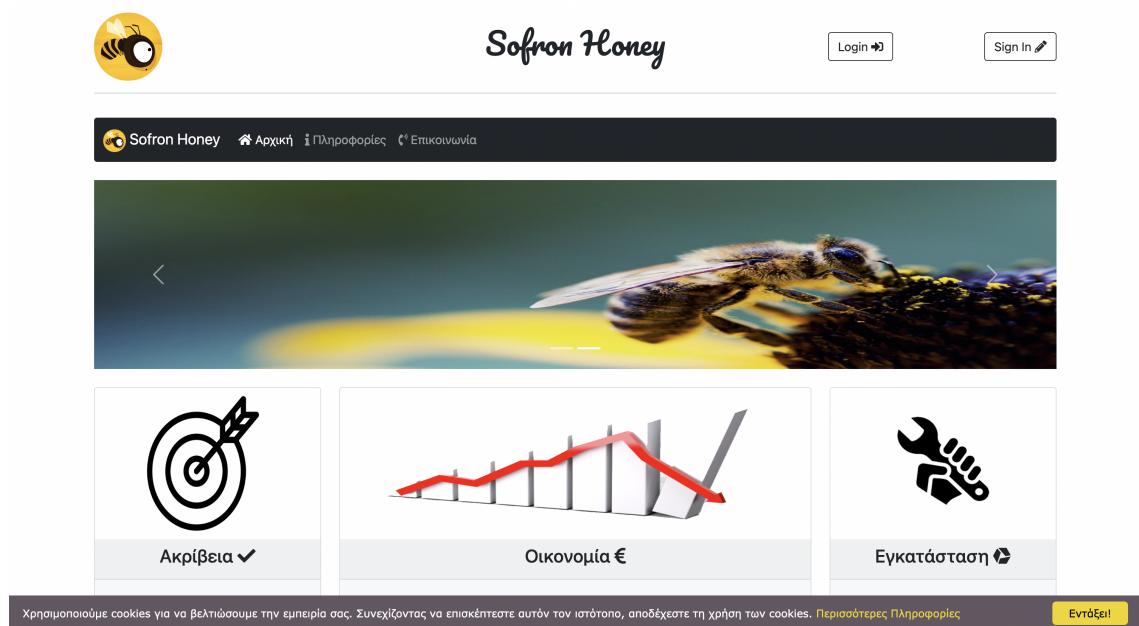
⁶Το αρχείο περιέχει σε κάθε γραμμή μια τυχαία σειρά από δεδομένα. Ο πρώτος αριθμός αντιστοιχίζεται στο πρωτεύον κλειδί, το οποίο αυξάνει τη τιμή του αυτομάτα, για αυτό η τιμή αυτή είναι μηδέν. Το όνομα, αποτελεί το όνομα του χρήστη που έχει αυτά τα δεδομένα. Η ημερομηνία είναι στη μορφή **έτος/μήνας/ημέρα** και ακολουθούν με τη σειρά η θερμοκρασία, η υγρασία και τέλος το βάρος.

				id	user	imerominia	thermokrasia	ugrasia	baros
<input type="checkbox"/>				1	sof	2022-01-01	1.2	22.9	39.57
<input type="checkbox"/>				2	sof	2022-01-02	23.26	12.4	158.04
<input type="checkbox"/>				3	sof	2022-01-03	28.53	59.27	65.22
<input type="checkbox"/>				4	sof	2022-01-04	18.41	71.94	45.68
<input type="checkbox"/>				5	sof	2022-01-05	11.57	67.94	21.73
<input type="checkbox"/>				6	sof	2022-01-06	43.21	43.4	15.11
<input type="checkbox"/>				7	sof	2022-01-07	45.71	2.19	112.33
<input type="checkbox"/>				8	sof	2022-01-08	10.56	29.66	138.63
<input type="checkbox"/>				9	sof	2022-01-09	24.03	30.19	151.98
<input type="checkbox"/>				10	sof	2022-01-10	18.47	69.38	77.19
<input type="checkbox"/>				11	sof	2022-01-11	34.73	22.61	87.42
<input type="checkbox"/>				12	sof	2022-01-12	47.86	49.68	152.35
<input type="checkbox"/>				13	sof	2022-01-13	34.93	97.21	161.52
<input type="checkbox"/>				14	sof	2022-01-14	40.81	75.36	158.8
<input type="checkbox"/>				15	sof	2022-01-15	18.61	52.61	96.01
<input type="checkbox"/>				16	sof	2022-01-16	32.41	49.83	164.64
<input type="checkbox"/>				17	sof	2022-01-17	36.18	18.2	112.44
<input type="checkbox"/>				18	sof	2022-01-18	43.96	76.26	38.82
<input type="checkbox"/>				19	sof	2022-01-19	38.55	76.98	164.75
<input type="checkbox"/>				20	sof	2022-01-20	37.93	90.35	131.56
<input type="checkbox"/>				21	sof	2022-01-21	33.05	90.3	35.24
<input type="checkbox"/>				22	sof	2022-01-22	21.27	22.73	45.3
<input type="checkbox"/>				23	sof	2022-01-23	29.36	16.95	66.23
<input type="checkbox"/>				24	sof	2022-01-24	36.89	80.56	80.67
<input type="checkbox"/>				25	sof	2022-01-25	42.54	90.04	167.02

Σχήμα 54: Επιβεβαίωση Εισαγωγής Δεδομένων από το αρχείο .csv

5.2.3 Πρόσβαση Στην Ιστοσελίδα

Η ιστοσελίδα είναι προσβάσιμη από οποιοδήποτε περιηγητή, (Chrome, Firefox, Safari, Opera κτλπ) από τη διεύθυνση, <http://localhost/thesis/>. Η εμφάνιση της ιστοσελίδας μετά από επιτυχή φόρτωση παρουσιάζεται παρακάτω.:



Σχήμα 55: Κεντρική Σελίδα

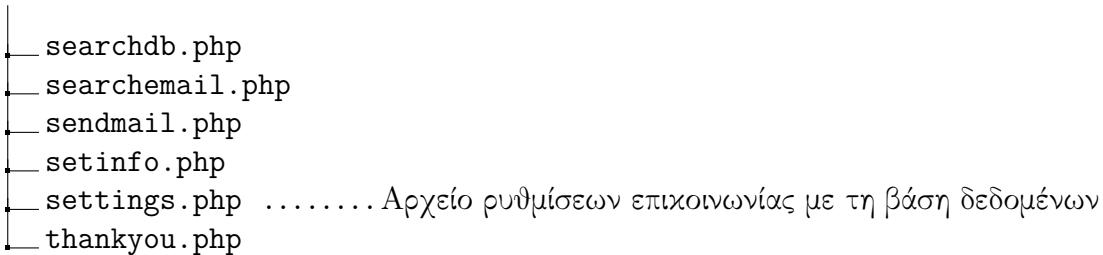
5.2.4 Δομή Καταλόγου

Για τη καλύτερη δυνατή οργάνωση της ιστοσελίδας, η δομή των αρχείων:

```

/
├── android .... Φάκελος με αρχεία που απαιτούνται από τη τηλεματική εφαρμογή
├── font ..... Φάκελος με αρχεία γραμματοσειρών
├── images ..... Φάκελος με εικόνες
├── js
└── managment ..... Περιβάλλον Προβολής Δεδομένων
    ├── css
    ├── img ..... Φάκελος με εικόνες
    ├── js
    ├── scss
    ├── vendor
    ├── getDayData.php
    ├── getMonthData.php
    ├── index.php ..... Αρχείο κεντρικής σελίδας δεδομένων
    ├── logtime.php
    ├── month.php
    ├── profile.php ..... Αρχείο προφίλ χρήστη
    ├── search.php
    ├── update_profile.php
    ├── week.php
    └── yesterday.php
├── check_login.php
├── checkuser.php
├── contact.php ..... Αρχείο φόρμας επικοινωνίας
├── error.php
├── forget.php
├── getinfo.php
├── getUser.php
├── index.php ..... Αρχείο κεντρικής σελίδας
├── info.php
├── insertuser.php
├── login.php ..... Αρχείο σύνδεσης χρήστη
├── logout.php ..... Αρχείο αποσύνδεσης χρήστη
├── logtime.php
├── register.php
└── registered.php

```



5.2.4.1 Αρχείο ρυθμίσεων σύνδεσης βάσης δεδομένων

- **settings.php**

Το αρχείο **settings.php** χρησιμοποιείται για την εύκολη αλλαγή βασικών στοιχείων σε περίπτωση μεταφοράς σε άλλο server, είτε μεταφοράς σε άλλη βάση δεδομένων, είτε αλλαγής στοιχείων σύνδεσης. Με αυτό το τρόπο, μπορεί να γίνει μια αλλαγή σε ένα μόνο αρχείο και όχι σε κάθε αρχείο που χρησιμοποιεί στοιχεία για τη σύνδεση με το σέρβερ και τη βάση δεδομένων.

```

1 <?php
2     const DB_HOST = 'localhost';
3     const DB_USER = 'root';
4     const DB_PASS = '';
5     const DB_NAME = 'android_application';
6 ?>
  
```

Στο συγκεκριμένο αρχείο ρυθμίσεων ορίζεται ως σέρβερ, ο τοπικός σέρβερ (localhost). Δηλώνεται το όνομα χρήστη, ο κωδικός πρόσβασης αλλά και το όνομα της βάσης δεδομένων.

5.2.4.2 Αρχείο κεντρικής σελίδας

Το αρχείο **index.php** βρίσκεται στο κεντρικό φάκελο με σκοπό να φορτώσει πρώτο, όταν ο χρήστης πλοηγηθεί στην σελίδα <http://localhost/thesis/>.

- **index.php**

Το αρχείο καλωσορίζει το χρήστη με την αρχική σελίδα, όταν προσπελάσει την ιστοσελίδα για πρώτη φορά. Η αρχική σελίδα ενημερώνει το χρήστη, για τις δυνατότητες που προσφέρει η ηλεκτρονική ζυγαριά. Από το μενού υπάρχουν οι **Πληροφορίες**, παρουσιάζονται τα υλικά που χρησιμοποιήθηκαν στην ηλεκτρονική ζυγαριά. Ακόμη υπάρχει διαθέσιμη φόρμα επικοινωνίας, μέσω της σελίδας, **Επικοινωνία**, για να επικοινωνήσει με τον διαχειριστή-προγραμματιστή.

Επίσης υπάρχει μενού επιλογής σύνδεσης, για τη σύνδεση του χρήστη στο σύστημα. Ο εκάστοτε χρήστης, έχει πρόσβαση στη προβολή των δεδομένων από την ηλεκτρονική ζυγαριά που αντιστοιχεί σε αυτόν. Υπάρχει η δυνατότητα εγγραφής του χρήστη στη πλατφόρμα, εάν δεν διαθέτει λογαριασμό.

Ακολουθεί επεξήγηση ανάπτυξης του θέματος. Η δομή ενός θέματος:

```

1 <!DOCTYPE html>
2 <html lang="el">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width,
6           initial-scale=1">
6       <title>Page Title </title>
7   </head>
8   <body>
9     <h1>Test text goes here.</h1>
10  </body>
11 </html>
```

Μια σελίδα χωρίζεται σε δύο μέρη, το ‘κεφάλι (**head**)’ και το ‘σώμα (**body**)’. Στο head ορίζονται όλα τα στοιχεία που πρέπει να φορτωθούν πριν το σώμα, συνήθως είναι βιβλιοθήκες ή αναγκαία αρχεία εμφάνισης (css), είτε αρχεία για λειτουργίες που προσφέρει η JavaScript.

Στο σώμα υλοποιείται η δομή και τα στοιχεία που προβάλλονται στο χρήστη. Η ιστοσελίδα χρησιμοποιεί δύο διαφορετικά θέματα. Ένα για το αρχικό περιβάλλον και ένα διαφορετικό θέμα για το διαχειριστικό περιβάλλον⁷. Αυτό έχει ως σκοπό, τη κατανόηση αλλαγής περιβάλλοντος από το χρήστη.

⁷Το δεύτερο θέμα υλοποιήθηκε και βρίσκεται στο φάκελο management.

The screenshot shows the Sofron Honey website. At the top right are 'Login' and 'Sign in' buttons. The main header 'Sofron Honey' is next to a yellow logo featuring a stylized bee. Below the header is a navigation bar with links: 'Sofron Honey', 'Αρχική', 'Πληροφορίες', and 'Επικοινωνία'. The main content area features a large image of a bee on a flower. To the left is a section titled 'Ακρίβεια' with a target icon and a red checkmark. Below it is a paragraph about accuracy. To the right is a section titled 'Οικονομία €' with a line graph icon and a red checkmark. Below it is a paragraph about cost-effectiveness. At the bottom right is a section titled 'Εγκατάσταση' with a wrench icon and a red checkmark. Below it is a paragraph about installation.

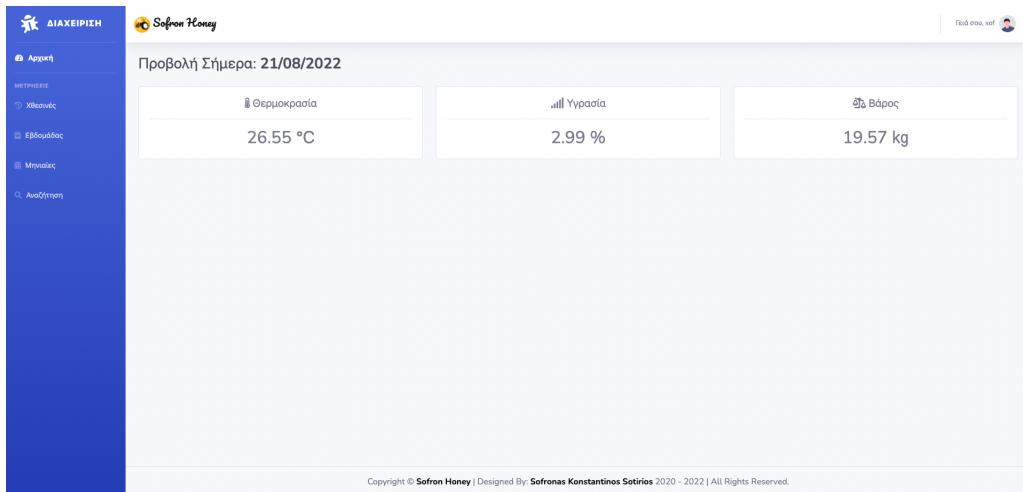
Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved.

Σχήμα 56: Θέμα Πρώτο

The screenshot shows the Sofron Honey website. At the top right is a dropdown menu showing 'Γειά σου, sof'. The main header 'Sofron Honey' is next to a yellow logo featuring a stylized bee. Below the header is a navigation bar with links: 'Sofron Honey', 'Αρχική', 'Πληροφορίες', and 'Επικοινωνία'. The main content area features a large image of a bee on a flower. To the left is a section titled 'Ακρίβεια' with a target icon and a red checkmark. Below it is a paragraph about accuracy. To the right is a section titled 'Οικονομία €' with a line graph icon and a red checkmark. Below it is a paragraph about cost-effectiveness. At the bottom right is a section titled 'Εγκατάσταση' with a wrench icon and a red checkmark. Below it is a paragraph about installation.

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved.

Σχήμα 57: Θέμα Πρώτο (Συνδεδεμένος Χρήστης)



Σχήμα 58: Θέμα Δεύτερο

5.2.4.3 Επεξήγηση δομής θέματος αρχικής σελίδας

Το πρώτο θέμα χωρίζεται σε τέσσερις περιοχές (containers). Κάθε μια απεικονίζει και μια συγκεκριμένη λειτουργία. Η πρώτη περιοχή, αφορά στο πάνω μέρος της σελίδας. Περιέχει τα λογότυπα της σελίδας στο αριστερό και μεσαίο μέρος ενώ στα δεξιά υπάρχουν δύο κουμπιά, ένα το οποίο όταν ο χρήστης το πατήσει θα τον οδηγήσει στη σελίδα σύνδεσης ([login.php](#)) και ένα άλλο που θα τον οδηγήσει στη σελίδα εγγραφής ([register.php](#)).



Σχήμα 59: Θέμα Πρώτο - Πρώτη περιοχή (container)

Αν ο χρήστης είναι συνδεδεμένος τότε η δεξιά πλευρά των κουμπιών από δύο αλλάζει σε ένα με ένα μήνυμα χαιρετισμού και το όνομα του χρήστη. Το κουμπί αυτό περιέχει ένα αναδυόμενο μενού με διάφορες επιλογές.



Σχήμα 60: (Εικόνα Αριστερά) - Θέμα Πρώτο - Πρώτη περιοχή (Συνδεδεμένος Χρήστης)

Σχήμα 61: (Εικόνα Δεξιά) - Θέμα Πρώτο - Πρώτη περιοχή (Μενού Χρήστη Αρχικής Σελίδας)

Η επιλογή **Διαχείριση**, οδηγεί τον χρήστη στο δεύτερο θέμα και περιβάλλον, **managment/index.php**, για τη προβολή των μετρήσεων. Η επιλογή **Προφίλ**, ο χρήστης παραμένει στο πρώτο θέμα και οδηγείται στη σελίδα **managment/profile.php** για τυχόν αλλαγές στα στοιχεία του. Ακόμη προσφέρεται επιλογή αποσύνδεσης του χρήστη από το σύστημα **logout.php**

Η δεύτερη περιοχή, αφορά τη μπάρα πλοήγησης (navigation) όπου ο χρήστης μπορεί να πλοηγηθεί στη σελίδα με τις πληροφορίες, **info.php**, αλλά και να χρησιμοποιήσει τη φόρμα επικοινωνίας που βρίσκεται στη σελίδα **contact.php**.



Σχήμα 62: Θέμα Πρώτο - Δεύτερη περιοχή (Μπάρα Πλοήγησης)

Στην τρίτη περιοχή, ο χρήστης βλέπει τη κεντρική περιοχή όπου βρίσκεται και το περιεχόμενο της σελίδας. Πρόκειται για μια παρουσίαση εικόνων και ένα πίνακα με τις δυνατότητες που προσφέρονται από την ηλεκτρονική ζυγαριά.





Ακρίβεια ✓

Η μελισσοκομική ζυγαριά χρησιμοποιεί αισθητήρες με ακρίβεια στις μετρήσεις του για καλύτερη διαφάλιση ενημέρωσης του μελισσοκόμου.



Οικονομία €

Ελαχιστοποιείστε τα έξοδα σας και τις συνεχές μετακινήσεις σας. Μειώστε σημαντικά τα έξοδα. Χρησιμοποιήστε την μελισσοκομική ζυγαριά για να παρακολουθήσετε την πορεία και την εξέλιξη της ανάπτυξης του μελισσοκομικού σας λαού.



Εγκατάσταση 🔧

Εύκολη εγκατάσταση στο χώρο του μελισσοκομικού σας λαού χωρίς να απαιτούνται γνώσεις.

Σχήμα 63: Θέμα Πρώτο - Τρίτη περιοχή (Περιεχόμενο σελίδας)

Τέλος, το θέμα κλείνει με τη τέταρτη περιοχή, η οποία αφορά τη παρουσίαση του κατόχου της ιστοσελίδας και συνήθως σε αυτή τη περιοχή γίνεται η απόδοση των απαραίτητων δικαιωμάτων. Αυτή η περιοχή είναι γνωστή ως footer.

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 – 2022 | All Rights Reserved.

Σχήμα 64: Θέμα Πρώτο - Τέταρτη περιοχή (Footer)

Η αρχική σελίδα υλοποιήθηκε μέσω του παρακάτω κώδικα:

```

1 <?php
2     //Initialize the session
3     session_start();
4
5     $inactive = 5; //seconds
6     if(isset($_SESSION['timeout'])) {
7         $session_life = time() - $_SESSION['timeout'];
8         if($session_life > $inactive)
9         {
10             session_destroy();
11             header("Location: Logout.php");
12         }
13     }
14     $_SESSION['timeout'] = time();
15 ?>
16
17 <!DOCTYPE html>
18 <html lang="el">
19     <head>
20         <meta charset="utf-8">
21         <meta name="viewport" content="width=device-width,
22         ↳ initial-scale=1">
23         <meta name="description" content="">
24         <meta name="author" content="Sofronas Konstantinos Sotirios">
25         <link rel="shortcut icon" href="images/fav.png"/>
26         <link href="css/sofron.css" rel="stylesheet">
27         <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css"
28         ↳ rel="stylesheet">
29             <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css"
27             ↳ rel="stylesheet">
28                 <link href="https://fonts.googleapis.com/css2?family=Pacifico&display=swap"
29                     ↳ rel="stylesheet">

```

```

30      <title>Sofron Honey</title>
31  </head>
32  <body>
33      <div class="container">
34          <!-- Top Logo Container -->
35          <div class="container">
36              <!-- Logo -->
37              <header class="blog-header py-3">
38                  <div class="row flexnowrap
39                      justify-content-between align-items-center">
40                      <div class="col">
41                          <a href="index.php"></a>
44                      </div>
45                      <div class="col text-center">
46                          <h1 class="blog-header-logo
47                             text-dark"><font face="Pacifico">Sofron Honey</font></h1>
48                      </div>
49                      <?php
50                          // Check if the user is logged in, if
51                          not then show page unlogged
52                          if(!isset($_SESSION["log"]) ||
53                             $_SESSION["log"] != true)
54                          {
55                              echo '<div class="col">
56                                  <div class="row"
57                                      id="buttonsTopRight">
58                                          <div class="col d-flex
59                                              justify-content-end">
60                                              <a type="button"
61                                              class="btn btn-outline-dark rounded" href="login.php">Login <i
62                                              class="fas fa-sign-in-alt"></i></a>
63                                          </div>
64                                      <div class="col d-flex
65                                          justify-content-end">

```

```

55                               <a type="button"
56   ↵   class="btn btn-outline-dark rounded" href="register.php">Sign In
57   ↵   <i class="fas fa-pencil-alt"></i></a>
58                               </div>
59                               </div>
60                               </div>';
61   }
62   elseif($_SESSION['log'] == TRUE)
63   {
64           echo '<div class="col"
65           <style="text-align:right">
66           <div class="dropdown
67           <show">
68           <a class="btn
69           btn-outline-dark dropdown-toggle" href="#" role="button"
70           id="dropdownMenuLink" data-bs-toggle="dropdown"
71           aria-expanded="false">
72               '. '<i
73               class="fas fa-user-alt"></i>&ampnbspΓειά σου, '. $_SESSION["user"]
74               .' <span class="caret"></span>
75               </a>
76           </div>
77           <div
78           class="dropdown-menu" aria-labelledby="dropdownMenuLink">
79               <a
80               class="dropdown-item" href="managment/index.php"><i class="fas
81               fa-tasks"></i>&ampnbspΔιαχείριση</a>
82               <a
83               class="dropdown-item" href="managment/profile.php"><i class="far
84               fa-id-card"></i>&ampnbspΠροφίλ</a>
85               <div
86               class="dropdown-divider"></div>
87               <a
88               class="dropdown-item" href="logout.php"><i class="fas
89               fa-sign-out-alt"></i>&ampnbspΑποσύνδεση</a>
90                   </div>
91               </div>
92           </div>
93       
```

```

76                                ' ;
77
78                            }
79                            else{
80                                echo "Something fucked them all...";
81                            }
82                        ?>
83
84                    </div>
85                    <hr>
86                </header>
87            </div>
88            <!-- Navigation Container -->
89            <div class="container">
90                <nav class="navbar navbar-expand-lg navbar-dark
91        bg-dark rounded">
92                    <div class="container-fluid">
93                        <a class="navbar-brand" href="index.php">&nbsp;Sofron
95                            Honey</a>
96                        <button class="navbar-toggler" type="button"
97                            data-bs-toggle="collapse"
98                            data-bs-target="#navbarSupportedContent"
99                            aria-controls="navbarSupportedContent" aria-expanded="false"
100                           aria-label="Toggle navigation">
101                            <span class="navbar-toggler-icon"></span>
102                        </button>
103                        <div class="collapse navbar-collapse"
104                            id="navbarSupportedContent">
105                            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
106                                <li class="nav-item">
107                                    <a class="nav-link active"
108                                       aria-current="page" href="index.php"><i class="fas fa-home"></i>
109                                       Αρχική</a>
110                                    </li>
111                                    <li class="nav-item">
112                                        <a class="nav-link" aria-current="page"
113                                           href="info.php"><i class="fas fa-info"></i> Πληροφορίες</a>
114                                    </li>

```

```

103                     <li class="nav-item">
104                         <a class="nav-link" aria-current="page"
105                             ↵ href="contact.php"><i class="fas fa-phone-volume"></i>
106                             ↵ Επικοινωνία</a>
107                     </li>
108                 </div>
109             </nav>
110         </div>
111         <br>
112         <!-- Body Container -->
113         <div class="container">
114             <div class="row">
115                 <div id="carouselExampleControls" class="carousel
116                   slide" data-bs-ride="carousel">
117                     <div class="carousel-indicators">
118                         <button type="button"
119                             ↵ data-bs-target="#carouselExampleIndicators" data-bs-slide-to="0"
120                             ↵ class="active" aria-current="true" aria-label="Slide 1"></button>
121                         <button type="button"
122                             ↵ data-bs-target="#carouselExampleIndicators" data-bs-slide-to="1"
123                             ↵ aria-label="Slide 2"></button>
124                     </div>
125                     <div class="carousel-inner">
126                         <div class="carousel-item active">
127                             
130                             </div>
131                         <div class="carousel-item">
132                             
135                             </div>
136                         </div>
137                         <a class="carousel-control-prev"
138                             ↵ href="#carouselExampleControls" type="button"
139                             ↵ data-bs-target="#carouselExampleControls" data-bs-slide="prev">
```

```

128                      <span class="carousel-control-prev-icon"
129             ↵  aria-hidden="true"></span>
130                  <span
131                     ↵  class="visually-hidden">Προηγούμενο</span>
132                     <a class="carousel-control-next"
133                        ↵  href="#carouselExampleControls" type="button"
134                           ↵  data-bs-target="#carouselExampleControls" data-bs-slide="next">
135                               <span class="carousel-control-next-icon"
136                                 ↵  aria-hidden="true"></span>
137                                 <span
138                                    ↵  class="visually-hidden">Επόμενο</span>
139
140                         </a>
141                         </div>
142                     </div>
143                     <br>
144                     <div class="row">
145                         <div class="col">
146                             <div class="card bg-light mb-3">
147                               
149                               <div class="card-header"
150                                 ↵  style="text-align:center">
151                                     <h4>Ακρίβεια <i class="fas
152                                       fa-check"></i></h4>
153
154                         </div>
155                         <div class="card-body">
156                             <p class="card-text"
157                               ↵  style="text-align:center">Η μελισσοκομική ζυγαριά χρησιμοποιεί
158                               ↵  αισθητήρες με ακρίβεια στις μετρήσεις του για καλύτερη διασφάλιση
159                               ↵  ενημέρωσης του μελισσοκόμου.</p>
160
161                         </div>
162                         </div>
163                     </div>
164                     <div class="col-md-6">
165                         <div class="card bg-light mb-3">
166                           

```

```

153                               <div class="card-header"
154             ↵ style="text-align:center">
155                     <h4>Οικονομία <i class="fas
156             ↵ fa-euro-sign"></i></h4>
157                     </div>
158             ↵ <div class="card-body">
159                     <p class="card-text"
160             ↵ style="text-align:center">Ελαχιστοποιείστε τα έξοδα σας και τις
161             ↵ συνεχές μετακινήσεις σας. Μειώστε σημαντικά τα έξοδα.
162             ↵ Χρησιμοποιήστε την μελισσοκομική ζυγαριά για να παρακολουθήσετε
163             ↵ την πορεία και την εξέλιξη της ανάπτυξης του μελισσοκομικού σας
164             ↵ λαού.</p>
165                     </div>
166             ↵ </div>
167         </div>
168             <div class="col">
169                     <div class="card bg-light mb-3">
170                         
172                         <div class="card-header"
173             ↵ style="text-align:center">
174                     <h4>Εγκατάσταση <i class="fab
175             ↵ fa-instalod"></i></h4>
176                     </div>
177                     <div class="card-body">
178                         <p class="card-text"
179             ↵ style="text-align:center">Εύκολη εγκατάσταση στο χώρο του
180             ↵ μελισσοκομικού σας λαού χωρίς να απαιτούνται γνώσεις.</p>
181                     </div>
182             ↵ </div>
183         </div>
184             <br>
185         </div>
186             <!-- Footer Container -->
187             <div class="container">
188                 <footer class="py-5">
189                     <hr>

```

```

179                     <div class="text-center">
180                         <p style="text-align: center">Copyright
181                         ← &copy;
182                         | <b><a class="no_a" href="index.php"
183                         | target="_blank">Sofron Honey</a></b>
184                         | Designed By: <a class="no_a"
185                         | href="https://sofronas.github.io/" target="_blank"><b>Sofronas
186                         | Konstantinos Sotirios</b></a>
187                         | 2020 - <?php echo date('Y'); ?> / All
188                         | Rights Reserved.
189                         | </p>
190                         | <button class="btn btn-success rounded"
191                         | onclick="topFunction()" id="gotopbutton" title="Πηγαίνε με στην
192                         | αρχή"><i class="fas fa-arrow-circle-up"></i></button>
193                         | </div>
194                         | </footer>
195                         | </div>
196                         | </div>
197                         | <!-- Javascript Session -->
198                         | <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"></script>
199                         | <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
200                         | <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"></script>
201                         | <!-- <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
202                         | <!-->
203                         | <script src="js/gotop.js"></script>
204
205                         | <!-- Cookies -->
206                         | <script type="text/javascript" id="cookieinfo"
207                         | src="//cookieinfoscript.com/js/cookieinfo.min.js"
208                         | data-bg="#645862" data-fg="#FFFFFF"

```

```
199      data-message="Χρησιμοποιούμε cookies για να βελτιώσουμε την  
↪ εμπειρία σας. Συνεχίζοντας να επισκέπτεστε αυτόν τον ιστότοπο,  
↪ αποδέχεστε τη χρήση των cookies. " data-linkmsg="Περισσότερες  
↪ Πληροφορίες" data-link="#F1D600" data-cookie="CookieInfoScript"  
↪ data-text-align="left" data-close-text="Εντάξει !">></script>  
200      <script type="text/javascript"  
↪ src="js/logoutsystem.js"></script>  
201      </body>  
202  </html>  
203
```

5.2.4.4 Αρχείο Πληροφοριών

- [info.php](#)

Το αρχείο πληροφοριών περιέχει αναφορικά τα υλικά που χρησιμοποιήθηκαν στην ηλεκτρονική ζυγαριά για την ενημέρωση του χρήστη.

The screenshot shows the Sofron Honey website's "Πληροφορίες" (Information) page. At the top, there is a navigation bar with a logo, the site name "Sofron Honey", and links for "Login" and "Sign In". Below the navigation bar, the page title "Πληροφορίες" is displayed. A note below the title states: "Τι χρησιμοποιήθηκε στην ηλεκτρονική ζυγαριά:" (What was used in the electronic scale?). Below this, there are eight items, each with an image and a caption:

- Arduino Uno Rev 3**: Ο "εγκέφαλος" της ηλεκτρονικής ζυγαριάς.
- Arduino GSM Shield**: Ο "μεσολαβητής".
- DHT11**: Αισθητήρας Γυρασίας.
- DS18B20**: Αισθητήρας Θερμοκρασίας.
- 150kg**: Αισθητήρας βάρους.
- Battery Adaptor**: Μετατρόπευκα για την κατάλληλη τροφοδοσία του Arduino.
- 12Volt 7.2Ah Battery**: Η πηγή ενέργειας της ηλεκτρονικής ζυγαριάς.
- Solar Panel 15W**: Ήλιακό Πάνελ 15W.
- Solar Charger**: Φορτιστής Μπαταρίας μέσω Ήλιακού Πάνελ.

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved.

Σχήμα 65: Σελίδα Πληροφοριών

5.2.4.5 Αρχείο Επικοινωνίας

- **contact.php**

Το αρχείο επικοινωνίας, περιέχει μια φόρμα στην οποία ο χρήστης μπορεί να έρθει σε επικοινωνία με το διαχειριστή της ιστοσελίδας, για τυχόν απορίες. Ο χρήστης πρέπει να συμπληρώσει όλα τα απαραίτητα πεδία.

Μετά το πάτημα του κουμπιού **Αποστολή**, ενεργοποιείται ο αλγόριθμος ελέγχου, ώστε να γίνει ο απαραίτητος έλεγχος των πεδίων και η αποστολή του μηνύματος, email, στο διαχειριστή μέσω του αρχείου **sendmail.php**.

The screenshot shows the Sofron Honey website's contact page. At the top, there is a logo of a bee inside a yellow circle, the text "Sofron Honey", and two buttons: "Login" and "Sign In". Below the header, there is a navigation bar with links for "Αρχική", "Πληροφορίες", and "Επικοινωνία". The main content area has a title "Φόρμα Επικοινωνίας" with a magnifying glass icon. It contains three input fields: one for the "Email address", one for "Το όνομα σας:", and one for "Το κείμενο σας:". Below these fields is a green "Αποστολή" button with a white arrow icon. At the bottom of the page, there is a copyright notice: "Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved."

Σχήμα 66: Σελίδα Επικοινωνίας

```
1 <script type="text/javascript">
2     $('#contactform').submit(function(e){
3         e.preventDefault();
4
5         var user_email =
6             document.getElementById('userEmail').value;
7         var user_name =
8             document.getElementById('userName').value;
9         var user_text =
10            document.getElementById('userText').value;
11
12         if((user_email == "") || (user_email == null)){
13             alert("Το email λείπει!");
14             return false;
15         }
16         if((user_name == "") || (user_name == null)){
17             alert("Το όνομα λείπει!");
18             return false;
19         }
20         if((user_text == "") || (user_text == null)){
21             alert("Το μήνυμα λείπει!");
22             return false;
23         }
24         var str = "sendmail.php?email=" + user_email +
25             "&name=" + user_name + "&message=" + user_text;
26         window.location.replace(str);
27     });
28 </script>
```

- **sendmail.php**

Η σελίδα αυτή είναι υπεύθυνη για την αποστολή του email στο διαχειριστή του site. Δέχεται τα στοιχεία, όνομα, email, από τη φόρμα επικοινωνίας και συντάσσει ένα μήνυμα. Η αποστολή του γίνεται χρησιμοποιώντας τον smtp server (Mercury) του Xampp.

Σε περίπτωση που η φόρμα επικοινωνίας, για οποιοδήποτε λόγο, στείλει ελλιπή στοιχεία, τότε καλείται να προβληθεί η σελίδα **error.php**. Άλλιώς, καλείται μια σελίδα με ευχαριστήριο μήνυμα προς το χρήστη μέσω της σελίδας **thankyou.php**.

```

1 <?php
2     if (empty($_GET)) {
3         header("Location: error.php");
4         die();
5     } else {
6         $to = "owner_email@mail.com";
7         $from = $_GET['email'];
8         $name = $_GET['name'];
9         $subject = "Sofron Honey | Form submission";
10        $message = $name . " wrote the following:" . "\n\n" .
11        $_GET['message'];
12
13        $headers = "From:" . $from;
14        mail($to, $subject, $message, $headers);
15        header("Location: thankyou.php");
16        die();
17    }
?>
```

- [thankyou.php](#)

The screenshot shows a website with a yellow header featuring a bee icon. The main title is "Sofron Honey". On the right, there are "Login" and "Sign In" buttons. Below the header, a dark navigation bar contains the text "Sofron Honey", "Αρχική", "Πληροφορίες", and "Επικοινωνία". The main content area has a large yellow banner with the text "Σας ευχαριστούμε!" and a smaller note below it. A blue button at the bottom says "Συνέχεια στην Αρχική".

Copyright © **Sofron Honey** | Designed By: **Sofronas Konstantinos Sotirios** 2020 - 2022 | All Rights Reserved.

Σχήμα 67: [thankyou.php](#) - Σελίδα Ευχαριστιών

- [error.php](#)⁸

The screenshot shows a website with a yellow header featuring a bee icon. The main title is "Sofron Honey". On the right, there are "Login" and "Sign In" buttons. Below the header, a dark navigation bar contains the text "Sofron Honey", "Αρχική", "Πληροφορίες", and "Επικοινωνία". The main content area has a large yellow banner with the text "Κάτι πήγε στραβά!" and a smaller note below it. A blue button at the bottom says "Συνέχεια στην Αρχική".

Copyright © **Sofron Honey** | Designed By: **Sofronas Konstantinos Sotirios** 2020 - 2022 | All Rights Reserved.

Σχήμα 68: [error.php](#) - Σελίδα Σφάλματος

⁸Οι σελίδες [thankyou.php](#) και [error.php](#) είναι ίδιες με τη διαφορά τους να είναι στο κειμένο.

5.2.4.6 Αρχείο Εγγραφής Χρήστη

- [register.php](#)
- [registered.php](#)

Προσφέρεται η δυνανότητα στο χρήστη να εγγραφεί στο σύστημα μέσω της φόρμας εγγραφών. Η φόρμα, περιέχει στοιχεία σχετικά με το όνομα, το επώνυμο, το όνομα χρήστη (username), το email, το κωδικό πρόσβασης και τέλος τον αριθμό κινητού που αντιστοιχεί στην ηλεκτρονική ζυγαριά που κατέχει.

Γίνεται έλεγχος στο σύστημα για διπλότυπες εγγραφές. Ο έλεγχος αφορά το όνομα χρήστη (username), τον αριθμό κινητού και το email για την αποφυγή διπλότυπων εγγραφών. Ο χρήστης ενημερώνεται με κατάλληλο μήνυμα.

Όνομα

Επώνυμο

Username

Email
 @

Κωδικός Πρόσβασης

Προβολή Κωδικού

Κινητό

Συμφωνώ [με τους όρους χρήστης](#).

Εγγραφή

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 – 2022 | All Rights Reserved.

Σχήμα 69: register.php - Σελίδα Εγγραφής

```
1 <?php
2     session_start();
3     if (isset($_SESSION['log'])){
4         if($_SESSION['log'] == TRUE){
5             header("Location: index.php");
6         }
7     }
8 ?>
9 <html lang="el">
10    <head>
11        <meta charset="utf-8">
12        <meta name="viewport" content="width=device-width,
13            initial-scale=1">
14        <meta name="description" content="">
15        <meta name="author" content="Sofronas Konstantinos Sotirios">
16        <link rel="shortcut icon" href="images/fav.png"/>
17        <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome |
18            /5.15.4/css/all.min.css"
19            rel="stylesheet">
20            <link href="https://fonts.googleapis.com/css?family=Robot |
21                o:300,400&display=swap"
22                rel="stylesheet">
23                <link href="https://apis.com/css2?family=Pacifico&display=swap"
24                    rel="stylesheet">
25                    <link rel="stylesheet" href="css/style.css">
26                    <link rel="stylesheet" href="css/sofron.css">
27                    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist |
28                        /css/bootstrap.min.css"
29                        rel="stylesheet">
```

22

```
23     <title>Register | Sofron Honey</title>
24
25     <style type="text/css">
26         .vs_ks{
27             visibility: hidden;
28         }
29     </style>
30   </head>
31   <body style="background-color: #eee;">
32       <section class="vh-100" >
33           <div class="container h-100">
34               <div class="row d-flex justify-content-center
35                   align-items-center h-100">
36                   <div class="col-lg-12 col-xl-11">
37                       <div class="card text-black" style="border-radius:
38                           25px;">
39                           <div class="card-body p-md-5">
40                               <div class="container">
41                                   <div class="row">
42                                       <div class="col"><a href="index.php"></a></div>
44                                       <div class="col align-items-center">
45                                           <h1 style="text-align:right;"><font
46                                               face="Pacifico">Sofron Honey</font></h1>
47                                           </div>
48                                       </div>
49                               <div class="container">
```

```

50          <div class="col-md-10 col-lg-6 col-xl-5 order-2
51          ↵  order-lg-1">
52
52          <p class="text-center h1 fw-bold mb-5 mx-1
53          ↵  mx-md-4 mt-4">Εγγραφή Χρήστη</p>
53          <form class="row g-3 needs-validation"
54          ↵  novalidate method="post" id="form_register" name="form_register">
54          <div class="col-md-12">
55          <label for="name"
55          ↵  class="form-label">Όνομα</label>
56          <input type="text" class="form-control"
56          ↵  id="name_id" value="" required>
57          <div class="invalid-feedback vs_ks"
57          ↵  id="name_id_error">
58          Συμπλήρωσε το όνομα
59          </div>
60          </div>
61          <div class="col-md-12">
62          <label for="surname"
62          ↵  class="form-label">Επώνυμο</label>
63          <input type="text" class="form-control"
63          ↵  id="surname_id" value="" required>
64          <div class="invalid-feedback vs_ks"
64          ↵  id="surname_id_error">
65          Συμπλήρωσε το επώνυμο
66          </div>
67          </div>
68          <div class="col-md-12">
69          <label for="username"
69          ↵  class="form-label">Username</label>
70          <div class="input-group has-validation">
```

```

71                      <input type="text" class="form-control"
72      ↵  id="username_id" required>
73                  <div class="invalid-feedback vs_ks"
74      ↵  id="username_id_error">
75                  Διαλεξε όνομα χρήστη
76                  </div>
77              </div>
78          <div class="col-md-12">
79              <label for="email"
80      ↵  class="form-label">Email</label>
81                  <div class="input-group has-validation">
82                      <span class="input-group-text">@</span>
83                      <input type="text" class="form-control"
84      ↵  id="email_id" required>
85                  <div class="invalid-feedback vs_ks"
86      ↵  id="email_id_error">
87                  Συμπλήρωσε το email
88                  </div>
89              </div>
90          <div class="col-md-12">
91              <label for="password"
92      ↵  class="form-label">Κωδικός Πρόσβασης</label>
93                  <input type="password" class="form-control"
94      ↵  id="password_id" required>
95                  <input class="form-check-input"
96      ↵  type="checkbox" onclick="togglerPass()">
97                  Προβολή Κωδικού
98                  <div class="invalid-feedback vs_ks"
99      ↵  id="password_id_error">
100                 Συμπλήρωσε κωδικό

```

```

94                     </div>
95                 </div>
96             <div class="col-md-12">
97                 <label for="phone"
98                     class="form-label">Κινητό</label>
99                     <input type="text" class="form-control"
100                    id="phone_id" maxlength="10" oninput="this.value =
101                      this.value.replace(/[^0-9.]/g, ' ').replace(/(\..*)\./g, '$1');"
102                      required>
103                     <div class="invalid-feedback vs_ks"
104                      id="phone_id_error">
105                         Συμπλήρωσε το κινητό
106                         </div>
107                     </div>
108                 <div class="col-12">
109                     <div class="form-check">
110                         <input class="form-check-input"
111                           type="checkbox" value="" id="invalidCheck" required>
112                         <label class="form-check-label"
113                           for="invalidCheck">
114                             Συμφωνώ <a href="#" data-toggle="modal"
115                               data-target="#terms">με τους όρους χρήσης.</a>
116                             </label>
117                         <div class="invalid-feedback vs_ks"
118                           id="terms_id_error">
119                             Πρέπει να συμφωνήσεις πρώτα με τους όρους
120                             χρήσης.
121                         </div>
122                     </div>
123                 </div>
124             <div class="col-12">

```

```

115                      <button class="btn btn-warning" type="button"
116      ↵  id="form_submit" onclick="Validate();">Εγγραφή <i class="fas
117      ↵  fa-signature"></i></button>
118
119          </div>
120
121          <!--  -->
124          
126
127          </div>
128
129          </div>
130          <div class="container">
131              <hr>
132
133          </div>
134
135          <div class="text-center">
136              <p style="text-align: center">Copyright &copy;
137
138              <b><a href="index.php" target="_blank">Sofron
139              ↵  Honey</a></b>
140
141                  | Designed By: <a
142
143                  ↵  href="https://sofronas.github.io/" target="_blank">Sofronas
144
145                  ↵  Konstantinos Sotirios</a>
146
147                  2020 - <?php echo date('Y'); ?> / All Rights
148
149                  ↵  Reserved.
150
151                  </p>
152
153          </div>

```

```
137          </footer>
138      </div>
139      </div>
140      </div>
141      </div>
142  </div>
143</section>
144
145  <!-- Javascript -->
146  <script src="js/jquery-3.3.1.min.js"></script>
147  <script src="js/popper.min.js"></script>
148  <script src="js/bootstrap.min.js"></script>
149  <script src="js/main.js"></script>
150  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist
151  ↵ /js/bootstrap.bundle.min.js"></script>
152
153  <!-- Modal Terms Section -->
154  <div id="terms" class="modal fade" role="dialog">
155    <div class="modal-dialog modal-lg">
156      <!-- Modal content-->
157      <div class="modal-content">
158        <div class="modal-header">
159          <button type="button" class="close btn btn-danger"
160  ↵ data-dismiss="modal">&times;</button>
161          <h4 class="modal-title">'Οροι Χρήσης</h4>
162        </div>
        <div class="modal-body">
```

163 <p>Ο παρών ιστοχώρος αναγνωρίζει και σέβεται το δικαίωμα
 ↳ προστασίας των προσωπικών δεδομένων των χρηστών του διαδικτύου,
 ↳ σύμφωνα με τον Ευρωπαϊκό Κανονισμό 2016/679. Το παρόν κείμενο
 ↳ πολιτικής περιγράφει τη συλλογή και επεξεργασία δεδομένων των
 ↳ χρηστών κατά την επίσκεψη ή χρήση των παρεχόμενων υπηρεσιών του
 ↳ Ιστοχώρου μας. Με την επίσκεψη και πλοήγησή σας στον ιστοχώρο
 ↳ μας, αποδέχεστε τους όρους που περιγράφονται στην πολιτική αυτή.

164

165 Ο επισκέπτης του Ιστοχώρου μας ή/και χρήστης των
 ↳ υπηρεσιών μας οφείλει να διαβάσει προσεκτικά τους όρους και τις
 ↳ προϋποθέσεις προστασίας των προσωπικών του πληροφοριών που
 ↳ αναφέρονται εδώ. Ενδέχεται να προβούμε σε τροποποίηση των όρων
 ↳ προστασίας των προσωπικών σας δεδομένων. Για αυτό παρακαλούμε, να
 ↳ ελέγχετε τακτικά τους εν λόγω όρους για τυχόν αλλαγές, καθ', όσον
 ↳ με την επίσκεψή σας αποδέχεστε τους παρόντες όρους και τις
 ↳ τροποποιήσεις τους.

166

167 Τι είναι δεδομένα προσωπικού χαρακτήρα

168

169 Σύμφωνα με τον νόμο, είναι κάθε πληροφορία που αφορά
 ↳ ταυτοποιημένο ή ταυτοποιήσιμο φυσικό πρόσωπο («υποκείμενο των
 ↳ δεδομένων»), όπως είναι ονοματεπώνυμο, ηλεκτρονική διεύθυνση, ΑΦΜ
 ↳ κ.ά. Ως ταυτοποιήσιμο ορίζεται το φυσικό πρόσωπο εκείνο του
 ↳ οποίου η ταυτότητα μπορεί να εξακριβωθεί, άμεσα ή έμμεσα, ιδίως
 ↳ μέσω αναφοράς σε αναγνωριστικό στοιχείο ταυτότητας, όπως όνομα,
 ↳ σε αριθμό ταυτότητας, σε δεδομένα θέσης, σε επιγραμμικό
 ↳ αναγνωριστικό ταυτότητας (διεύθυνση IP, e-mail κλπ.) ή σε έναν ή
 ↳ περισσότερους παράγοντες που προσιδιάζουν στη σωματική,
 ↳ φυσιολογική, γενετική, ψυχολογική, οικονομική, πολιτιστική ή
 ↳ κοινωνική ταυτότητά του.

170

171 Τι είναι επεξεργασία προσωπικών δεδομένων

- 172

- 173 Επεξεργασία δεδομένων προσωπικού χαρακτήρα είναι κάθε
- ↳ πράξη ή σειρά πράξεων που πραγματοποιείται με ή χωρίς τη χρήση
- ↳ αυτοματοποιημένων μέσων, σε δεδομένα προσωπικού χαρακτήρα, όπως η
- ↳ συλλογή, καταχώρηση, οργάνωση, διάρθρωση, αποθήκευση, προσαρμογή,
- ↳ μεταβολή, ανάκτηση, αναζήτηση πληροφοριών, χρήση, διαβίβαση σε
- ↳ τρίτους, διάδοση, συσχέτιση, συνδυασμός, περιορισμός, διαγραφή
- ↳ και καταστροφή προσωπικών δεδομένων.
- 174

- 175 Ποια προσωπικά δεδομένα συλλέγουμε και για ποιο σκοπό
- 176

- 177 Κατά την επίσκεψη και πλοήγηση στον Ιστοχώρο μας
- ↳ συλλέγουμε τα παρακάτω προσωπικά δεδομένα:
- 178

- 179 α) δεδομένα πλοήγησης
- 180

- 181 Ο ιστοχώρος μας συγκεντρώνει στοιχεία αναγνώρισης των
- ↳ χρηστών του χρησιμοποιώντας τεχνολογίες, όπως τα cookies ή/και
- ↳ την παρακολούθηση διευθύνσεων Πρωτοκόλλου Internet (IP). Τα
- ↳ cookies είναι μικρά αρχεία κειμένου που αποθηκεύονται στο σκληρό
- ↳ δίσκο κάθε επισκέπτη/ χρήστη και δεν λαμβάνουν γνώση οποιουδήποτε
- ↳ εγγράφου ή αρχείου από τον υπολογιστή του. Επιπλέον, συλλέγονται
- ↳ αυτόματα πληροφορίες σχετικά με τις τοποθεσίες που επισκέπτεται ο
- ↳ επισκέπτης και σχετικά με τους συνδέσμους σε ιστοχώρους τρίτων
- ↳ που ενδέχεται να επιλέξει ο χρήστης. Η επεξεργασία των παραπάνω
- ↳ δεδομένων γίνεται με σκοπό να σας παρέχουμε απρόσκοπτη πρόσβαση
- ↳ στις πληροφορίες του Ιστοχώρου μας.
- 182

- 183 β) Φόρμα επικοινωνίας
- 184

- 185 Εφόσον χρησιμοποιείτε τη φόρμα επικοινωνίας μας για να
 ↪ επικοινωνήσετε με τον Ιστοχώρο μας, συλλέγουμε το όνομά σας, τη
 ↪ διεύθυνση ηλεκτρονικής αλληλογραφίας σας και το τηλέφωνό σας με
 ↪ τη συγκατάθεσή σας, ώστε να επικοινωνήσουμε μαζί σας.
- 186

- 187 Νόμιμη βάση επεξεργασίας των προσωπικών δεδομένων
- 188

- 189 Η νόμιμη βάση επεξεργασίας των προσωπικών δεδομένων είναι
 ↪ η συγκατάθεση κατά την επίσκεψη της ιστοσελίδας μας.
- 190

- 191 Αποδέκτες των δεδομένων
- 192

- 193 Τα προσωπικά δεδομένα των επισκεπτών ή/και χρηστών των
 ↪ υπηρεσιών του Ιστοχώρου μας δεν διαβιβάζονται σε τρίτους
 ↪ αποδέκτες, με εξαίρεση την περίπτωση όπου παρέχετε τη συγκατάθεσή
 ↪ σας για το σκοπό αυτό ή για τη θεμελίωση, άσκηση ή υποστήριξη
 ↪ νομικών αξιώσεων ή εάν αυτό είναι απαραίτητο για τη συμμόρφωση με
 ↪ έννομη υποχρέωσή μας.
- 194

- 195 Χρόνος διατήρησης των δεδομένων
- 196

- 197 Τα προσωπικά δεδομένα που αφορούν τους επισκέπτες του
 ↪ Ιστοχώρου ή/και χρήστες των ηλεκτρονικών μας υπηρεσιών
 ↪ συλλέγονται και διατηρούνται για τον απολύτως απαραίτητο χρόνο
 ↪ και μετά διαγράφονται.
- 198

- 199 Cookies
- 200

201 Κατά την πλοήγηση στον παρόντα ιστότοπο, όπως
 ↵ προαναφέρθηκε, ενδέχεται να συγκεντρώνονται στοιχεία αναγνώρισης
 ↵ των χρηστών με τη χρήση cookies. Η χρήση των cookies διευκολύνει
 ↵ τον ιστοχώρο μας να αποθηκεύει πληροφορίες σχετικά με την
 ↵ επίσκεψη του χρήστη, την ασφαλή αναζήτηση, τον υπολογισμό του
 ↵ αριθμού των επισκεπτών ή τη διευκόλυνση της εγγραφής στις
 ↵ υπηρεσίες μας.

202

203 Μπορείτε να διαγράψετε τα cookies, αν το επιλέξετε.

↪ Σχετικά δείτε εδώ: www.aboutcookies.org Μπορείτε να διαγράψετε
 ↵ όλα τα cookies που βρίσκονται ήδη στον υπολογιστή σας, όπως και
 ↵ να ρυθμίσετε τους περισσότερους browsers, ώστε που να μην
 ↵ επιτρέπουν την εγκατάσταση cookies. Ωστόσο, εάν επιλέξετε να
 ↵ διαγράψετε ή να καταργήσετε ορισμένα cookies, ενδέχεται να
 ↵ επηρεαστεί η λειτουργία του ιστοχώρου μας.

204

205

206 Ασφάλεια δεδομένων

207

208 Για την προστασία των προσωπικών δεδομένων των χρηστών

↪ και επισκεπτών του Ιστοχώρου μας χρησιμοποιούμε ασφαλή σύνδεση
 ↵ και μέτρα ασφαλείας δεδομένων για την αποτροπή του κινδύνου
 ↵ απώλειας, κακής χρήσης, μη εξουσιοδοτημένης πρόσβασης και
 ↵ κοινοποίησης των προσωπικών σας πληροφοριών.

209

210 Τα δικαιώματα σας

211

212 Όσον αφορά την προστασία των δεδομένων προσωπικού

↪ χαρακτήρα που σας αφορούν, έχετε τα εξής δικαιώματα:
 213 α) δικαίωμα πρόσβασης, δηλ. το δικαίωμα να πληροφορηθείτε
 ↵ εάν προσωπικά δεδομένα που σας αφορούν υφίστανται επεξεργασία,

214 β) δικαίωμα διόρθωσης, δηλ. το δικαίωμα να απαιτήσετε τη
 ↪ διόρθωση ανακριβών δεδομένων προσωπικού χαρακτήρα,
 215 γ) δικαίωμα διαγραφής, δηλ. το δικαίωμα να ζητήσετε να
 ↪ διαγραφούν τα προσωπικά δεδομένα που σας αφορούν,
 216 δ) δικαίωμα περιορισμού της επεξεργασίας όταν η ακρίβεια
 ↪ των δεδομένων αμφισβητείται, είναι παράνομη ή για άλλους λόγους
 217 ε) δικαίωμα φορητότητας, δηλ. το δικαίωμα να ζητήσετε να
 ↪ λάβετε τα δεδομένα που σας αφορούν σε δομημένο, κοινώς
 ↪ χρησιμοποιούμενο και αναγνώσιμο από μηχανήματα μορφότυπο, καθώς
 ↪ και να διαβιβασθούν τα εν λόγω δεδομένα σε άλλον υπεύθυνο
 ↪ επεξεργασίας και
 218 στ) δικαίωμα εναντίωσης στην επεξεργασία των προσωπικών
 ↪ δεδομένων σας, εκτός αν υφίστανται επιτακτικοί και νόμιμοι λόγοι
 ↪ για την επεξεργασία οι οποίοι υπερισχύουν των συμφερόντων, των
 ↪ δικαιωμάτων και των ελευθεριών σας.</p>

219 </div>
 220 </div>
 221 </div>
 222 </div>

223

224 <script type="text/javascript">
 225 function toggerPass(){
 226 var t = document.getElementById('password_id').type;
 227 if(t === "password") {
 228 document.getElementById('password_id').type = "text";
 229 }
 230 if(t === "text"){
 231 document.getElementById('password_id').type = "password";
 232 }
 233 }
 234 function Validate(){
 235 var email = document.getElementById('email_id').value;

```

236         var username =
237             ↵ document.getElementById('username_id').value;
238         var phone = document.getElementById('phone_id').value;
239         var name = document.getElementById('name_id').value;
240         var surname = document.getElementById('surname_id').value;
241         var password =
242             ↵ document.getElementById('password_id').value;
243         var terms =
244             ↵ document.getElementById('invalidCheck').checked;
245
246         var email_error =
247             ↵ document.getElementById('email_id_error');
248         var username_error =
249             ↵ document.getElementById('username_id_error');
250         var phone_error =
251             ↵ document.getElementById('phone_id_error');
252         var name_error = document.getElementById('name_id_error');
253         var surname_error =
254             ↵ document.getElementById('surname_id_error');
255         var password_error =
256             ↵ document.getElementById('password_id_error');
257         var terms_error =
258             ↵ document.getElementById('terms_id_error');

259
260         var ins = true;

261
262         if(email == ""){
263             //empty email
264             email_error.style.visibility = "visible";
265             ins = false;
266         }
267         if(username == ""){

```

```
259         //empty username
260         username_error.style.visibility = "visible";
261         ins = false;
262     }
263     if(phone == ""){
264         //empty phone
265         phone_error.style.visibility = "visible";
266         ins = false;
267     }
268     if(name == ""){
269         //empty name
270         name_error.style.visibility = "visible";
271         ins = false;
272     }
273     if(surname == ""){
274         //empty surname
275         surname_error.style.visibility = "visible";
276         ins = false;
277     }
278     if(password == ""){
279         //empty password
280         password_error.style.visibility = "visible";
281         ins = false;
282     }
283     if(terms == false){
284         //empty terms
285         terms_error.style.visibility = "visible";
286         ins = false;
287     }
288     var okay = false;
289     if(ins == true){
290         // call insertion fuction
```

```

291         okay = checkUserData(email,phone,username);
292         if(okay == true){
293             var k =
294             → insertNewUser(email,phone,username,name,surname,password);
295                 // happend error
296                 // alert("k= " + k);
297                 if(k == false){
298                     $('#data_error').modal('show');
299                 } else if(typeof(k) === 'undefined') {
300                     $('#data_error').modal('show');
301                 } else if(k == true) {
302                     var url = "registered.php";
303                     var link = url + "?username=" + username;
304                     window.location.replace(link);
305                 }
306                 } else {
307                     $('#data_exists').modal('show');
308                 }
309                 } else {
310                     $('#data_error').modal('show');
311                 }
312         function checkUserData(email,phone,username){
313             var http = new XMLHttpRequest();
314             var val;
315             http.onreadystatechange = function() {
316                 if (this.readyState == 4 && this.status == 200) {
317                     if(http.responseText >= 1){
318                         val = false;
319                     } else {
320                         // alert("checkUserData= true");
321                         val = true;

```

```

322         }
323     }
324     } ;
325     http.open("POST", "checkuser.php", false);
326     http.setRequestHeader("Content-Type",
327     ↪ "application/x-www-form-urlencoded");
328     http.send("username=" + username + "&phone=" + phone +
329     ↪ "&email=" + email);
330     return val;
331   }
332   function
333   ↪ insertNewUser(email, phone, username, name, surname, password){
334     var http = new XMLHttpRequest();
335     var x;
336     http.onreadystatechange = function() {
337       if (this.readyState == 4 && this.status == 200) {
338         // alert(http.responseText);
339         if(http.responseText == "email_exists"){
340           x = false;
341         } else if(http.responseText == "phone_exists"){
342           x = false;
343         }else if(http.responseText == "username_exists"){
344           x = false;
345         } else {
346           x = true;
347         }
348       }
349     };
350     http.open("POST", "insertuser.php", false);
351     http.setRequestHeader("Content-Type",
352     ↪ "application/x-www-form-urlencoded");

```

```
349         http.send("username=" + username + "&phone=" + phone +
350             "&email=" + email + "&name=" + name + "&surname=" + surname +
351             "&password=" + password);
352     return x;
353   }
354
355   <!-- Modal section -->
356
357   <div class="modal fade" id="data_error" tabindex="-1"
358       role="dialog" aria-labelledby="data_error" aria-hidden="true"
359       data-bs-backdrop="static" data-bs-keyboard="false"
360       style="text-align: center;">
361     <div class="modal-dialog modal-dialog-centered"
362       role="document">
363       <div class="modal-content">
364         <div class="modal-header">
365           <h5 class="modal-title">ΠΡΟΣΟΧΗ <i class="fas
366             fa-exclamation-triangle"></i></h5>
367         </div>
368         <div class="modal-body">
369           <h3>Δεν έχουν συμπληρωθεί όλα τα στοιχεία.</h3>
370         </div>
371         <div class="modal-footer">
372           <button type="button" class="btn btn-secondary"
373             data-bs-dismiss="modal">Κλείσιμο</button>
374         </div>
375       </div>
376     </div>
377   </div>
```

```

373      <div class="modal fade" id="data_exists" tabindex="-1"
374          ↵ role="dialog" aria-labelledby="data_exists" aria-hidden="true"
375          ↵ data-bs-backdrop="static" data-bs-keyboard="false"
376          ↵ style="text-align: center;">
377              <div class="modal-dialog modal-dialog-centered"
378                  ↵ role="document">
379                  <div class="modal-content">
380                      <div class="modal-header">
381                          <h5 class="modal-title">ΠΡΟΣΟΧΗ <i class="fas
382                          ↵ fa-exclamation-triangle"></i></h5>
383                      </div>
384                      <div class="modal-body">
385                          <h3>Τα στοιχεία που δήλωσες υπάρχουν ήδη.</h3>
386                      </div>
387                      <div class="modal-footer">
388                          <button type="button" class="btn btn-secondary"
389                          ↵ data-bs-dismiss="modal">Κλείσιμο</button>
390                      </div>
391                  </div>
392              </div>
393          </div>
394      </div>
395
396
397
398
399      </body>
400  </html>

```

Εφόσον ο χρήστης συμπληρώσει όλα τα πεδία της φόρμας και περάσει από όλους τους ελέγχους, τότε καθοδηγείται στη σελίδα ενημέρωσης (registered.php) της δημιουργίας του λογαριασμού, με όλα τα στοιχεία που καταχωρήθηκαν στο σύστημα. Εντός λίγων λεπτών, η σελίδα ανανεώνεται και ο χρήστης οδηγείται στην αρχική σελίδα. Ακόμη, αποστέλλεται ενημερωτικό email στο χρήστη, με όλα τα στοιχεία που δήλωσε κατά την εγγραφή του στο σύστημα.

Ακολουθεί παράδειγμα μετά από εγγραφή ενός χρήστη στο σύστημα:



Sofron Honey



Σας ευχαριστούμε για την εγγραφή σας!

#	Username	Τηλέφωνο	Όνομα	Επώνυμο	Email
2	test	6983733806	τεστ	τεστ	test@gmail.com

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved.

Σχήμα 70: registered.php - Σελίδα Επιτυχούς Εγγραφής Χρήστη

Εγγραφή στο site: Sofron Honey Εισερχόμενα ×



sofronas.sotiris@gmail.com

προς sofronas.konstantinos, κρυφή κοιν.: εγώ ▾

id	Username	Password	Τηλέφωνο	Όνομα	Επώνυμο	Email
1	sof	kostas	6983733807	Konstantinos	Sofronas	sofronas.konstantinos@gmail.com

Σχήμα 71: Email - Στοιχείων Εγγραφής Χρήστη

5.2.4.7 Αρχείο Σύνδεσης Χρήστη

- **login.php**

Το αρχείο ελέγχου της εισόδου του χρήστη στο σύστημα, (login.php), αποτελείται από μια φόρμα ελέγχου στοιχειών επιβεβαίωσης με σκοπό τη προβολή δεδομένων από την ηλεκτρονική ζυγαριά.

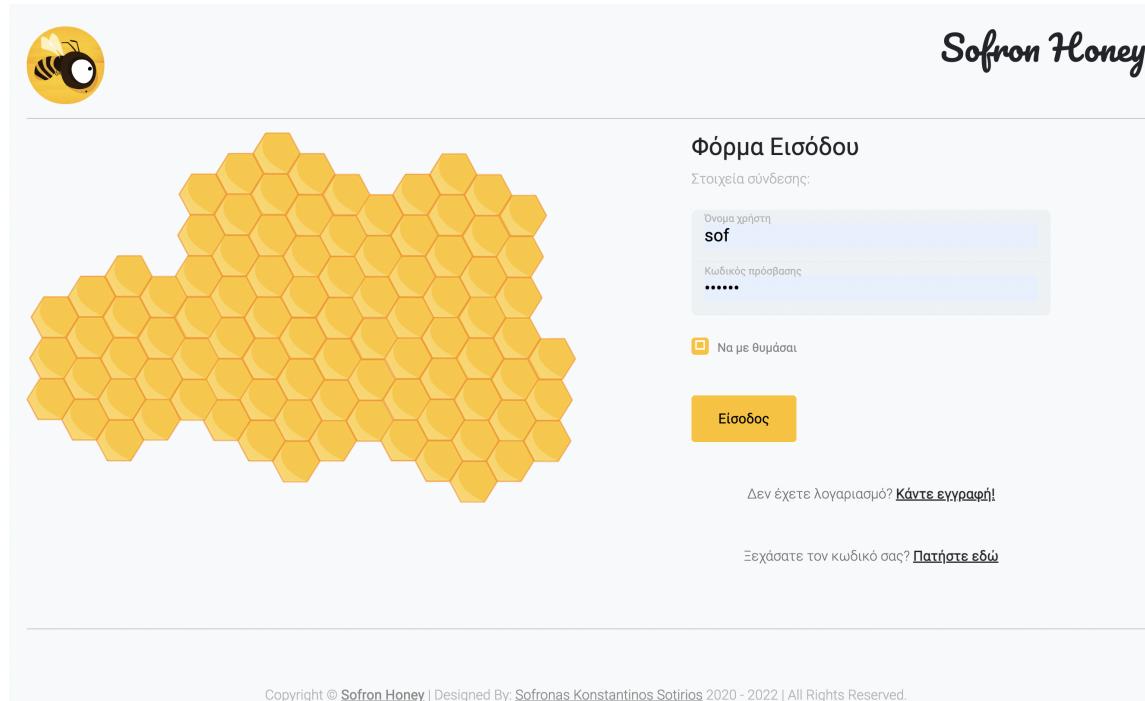
Πριν ξεκινήσει η φόρτωση του αρχείου, γίνεται έλεγχος αν ο χρήστης είναι ήδη συνδεδεμένος στο σύστημα μέσω των μεταβλητών του **`$_SESSION`** που προσφέρει ο πλοηγητής.

```

1 <?php
2     session_start();
3     if (isset($_SESSION['log'])) {
4         if ($_SESSION['log'] == TRUE) {
5             header("Location: index.php");
6         }
7     }
8 ?>

```

Στη περίπτωση που ο χρήστης είναι ήδη συνδεδεμένος στο σύστημα, τότε ανακατεύθυνεται στην αρχική σελίδα. Σε αντίθετη περίπτωση, κατευθύνεται στη φόρμα σύνδεσης η οποία φαίνεται παρακάτω.



Σχήμα 72: login.php - Σελίδα Σύνδεσης

Όταν το κουμπί **Είσοδος** πατηθεί από το χρήστη, τότε εκτελείται ο αλγόριθμος ελέγχου:

```

1 <script type="text/javascript">
2     function checkUser(){
3         var username = document.getElementById('username').
4             value;
5         var password = document.getElementById('password').
6             value;
7         var final_url = "getinfo.php";
8         var final_link = final_url + '?username=' + username +
9             '&password=' + password;
10
11        var url = "check_login.php";
12        var link = url + '?username=' + username + '&password='
13             + password;
14
15        var xmlhttp = new XMLHttpRequest();
16        xmlhttp.onreadystatechange = function() {
17
18            if(xmlhttp.readyState == 4 && xmlhttp.status == 200)
19            {
20                document.getElementById('msg').innerHTML =
21                    xmlhttp.responseText;
22            }
23        }
24    }
25
26    checkUser();
27
28</script>
```

```

13      if (xmlHttp.readyState == 4 && xmlHttp.status ==
200)
14      {
15          // alert(xmlHttp.responseText);
16          if(xmlHttp.responseText == "1"){
17              window.location.replace(final_link);
18          } else {
19              $('#wrong').modal('show');
20          }
21      }
22      // alert(link);
23      xmlhttp.open("GET", link, false); // true for
asynchronous
24      xmlhttp.send(null);
25
26
27  }
28 </script>

```

Γίνεται συλλογή των στοιχείων από τα πεδία που συμπλήρωσε ο χρήστης, δηλαδή το όνομα χρήστη και ο κωδικός πρόσβασης. Αποστέλλονται στο αρχείο **check_login.php** για γίνει ο απαιτούμενος έλεγχος.

- **check_login.php**

Το αρχείο καλείται από τη σελίδα **login.php**, με σκοπό την επαλήθευση ύπαρξης του χρήστη στη βάση δεδομένων. Η κλήση του αρχείου γίνεται με ασύχρονο τρόπο, δηλαδή η ενέργεια αναζήτησης γίνεται στο παρασκήνιο. Μόλις βρεθεί η απάντηση, ειδοποιείται η σελίδα για να προβεί στην ανάλογη καθορισμένη ενέργεια που έχει οριστεί.

```
1 <?php
2     require 'settings.php';
3
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     if (!$conn){
6         die("Connection failed: " . mysqli_connect_error());
7         echo "failed";
8     }
9
10    $username = $_GET['username'];
11    $password = $_GET['password'];
12    if ($username == NULL){
13        echo "empty_username";
14        mysqli_close($conn);
15        die();
16    }
17    if ($password == NULL){
18        echo "empty_password";
19        mysqli_close($conn);
20        die();
21    }
22    $sql = "SELECT * FROM users
23           WHERE username = '" . $username . "' AND password='"
24           . $password . "'";
25
26    $result = mysqli_query($conn, $sql);
27    $check = mysqli_num_rows($result);
28
29    if ($check > 0){
30        while ($data = mysqli_fetch_assoc($result)){
31            echo "1";
32        }
33    } else {
34        echo "3";
35    }
36
```

```

34    }
35
36     mysqli_close($conn);
37 ?>

```

Στην αρχή του αρχείου γίνεται η συμπεριληψη των ρυθμίσεων που υπάρχουν στο αρχείο **settings.php**, συνεχίζεται με σύνδεση στη βάση δεδομένων με χρήση της συνάρτηση, **mysqli()**.

Σε περίπτωση λανθασμένων στοιχείων, η σύνδεση τερματίζεται και επιστρέφεται μήνυμα λάθους ενημερώνοντας κατάλληλα το χρήστη. Σε αντίθετη περίπτωση, γίνεται η αρχικοποιήση των μεταβλητών που έστειλε η σελίδα **login.php**. Ακολουθεί η εύρεση του χρήστη, μέσω ερωτήματος, με χρήση της **SQL**, στη βάση δεδομένων.

Αν ο χρήστης βρεθεί, τότε γίνεται επιστροφή τιμής 1 αλλιώς η τιμή 3 επιστρέφεται. Η ένδειξη της τιμής 1 δηλώνει ότι ο χρήστης βρέθηκε ενώ η τιμή 3 δηλώνει ότι ο χρήστης δεν βρέθηκε.

Στη συνέχεια, εφόσον η σελίδα **login.php** λάβει απάντηση από τη **check_login.php**, 1 ή 3 για την ύπαρξη ή μη του χρήστη. Αν ο χρήστης βρεί το χρήστη καλεί την σελίδα **getinfo.php** με σκοπό την αρχικοποιήση της σύνδεσης του με τον server.

- [getinfo.php](#)

```

1 <?php
2     require 'settings.php';
3     session_start();
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     if (!$conn) {
6         die("Connection failed: " . mysqli_connect_error());
7         echo "failed";
8     }
9     $user = $_GET['username'];
10    $pass = $_GET['password'];
11    if (!isset($user) && !isset($pass))
12    {
13        $_SESSION['log'] = true;
14        $_SESSION['user'] = $user;
15    }
16    else {
17        $_SESSION['log'] = false;
18        mysqli_close($conn);
19        header("Location: index.php");
20        exit;
21    }
22    $sql = "SELECT * FROM users
23        WHERE username = '" . $user . "'"
24        AND password = '" . $pass . "'";
25    echo $sql;
26    $result = mysqli_query($conn, $sql);
27    $name = "";
28    $surname = "";
29    $phone = "";
30    $id = "";
31    $email = "";
32    if (mysqli_num_rows($result) > 0) {
33        while ($row = mysqli_fetch_row($result)) {
34            $user_username = $row[1];
35            $name = $row[5];
36            $surname = $row[6];
37            $email = $row[7];
38            $phone = $row[3];
39            $id = $row[0];

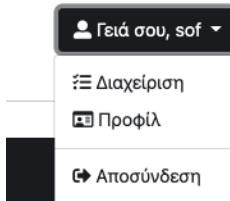
```

```

40    }
41    $_SESSION[ 'id' ] = $id ;
42    $_SESSION[ 'username' ] = $user_username ;
43    $_SESSION[ 'log' ] = true ;
44    $_SESSION[ 'user' ] = $user ;
45    $_SESSION[ 'name' ] = $name ;
46    $_SESSION[ 'surname' ] = $surname ;
47    $_SESSION[ 'phone' ] = $phone ;
48    $_SESSION[ 'email' ] = $email ;
49    $_SESSION[ 'timeout' ] = time() ;
50    header( "Location: index.php" );
51 } else {
52     echo '<script language="javascript">';
53     echo 'alert(Wrong data)';
54     echo '</script>';
55     header( "Location: index.php" );
56     exit;
57 }
58 mysqli_close($conn);
59 ?>
```

Η σελίδα **getinfo.php**, είναι υπεύθυνη για την αρχικοποιήση της συνεδρίας (**\$_SESSION**), για τη διατήρηση στοιχείων μεταξύ του χρήστη και του server. Εφόσον έχουν γίνει όλοι οι απαραίτητοι έλεγχοι, αρχίζει η ανάθεση τιμών σε μεταβλητές του τύπου της σύνδεσης, με τα στοιχεία του χρήστη από τη βάση δεδομένων.

Σε περίπτωση σφάλματος, ο χρήστης επιστρέφει στην αρχική σελίδα χωρίς να έχει γίνει αρχικοποίηση της συνεδρίας. Εάν γίνει η αρχικοποιήση τιμών, ο χρήστης επιστρέφει στην αρχική σελίδα, έχοντας αυτή την φορά στη προβολή θέματος ένα κουμπί με μήνυμα καλωσορίσματος και το όνομα του, αντί δύο κουμπιών, σύνδεσης και εγγραφής. Αυτό συμβαίνει καθώς ορίστηκε η τιμή **\$_SESSION['log'] = true;**.



Σχήμα 73: Αλλαγή σελίδας μετά τη σύνδεση του χρήστη

5.2.4.8 Επεξεργασία στοιχείων Προφίλ Χρήστη

- [managment/profile.php](#)

Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved.

Σχήμα 74: Προφίλ Χρήστη

Η σελίδα Προφίλ, [profile.php](#), περιέχει μια φόρμα με όλα τα στοιχεία του χρήστη. Το όνομα χρήστη (username), το όνομα, το επώνυμο, ο τηλεφωνικός αριθμός της ηλεκτρονικής ζυγαριάς αλλά και το email του. Υπάρχει η δυνατότητα αλλαγής, αυτών των στοιχείων [update_profile.php](#).

Η σελίδα [profile.php](#), διαβάζει τα πεδία από τη φόρμα όταν πατηθεί το κουμπί **Αποθήκευση Αλλαγών** και καλεί την σελίδα [update_profile.php](#), με τις αλλαγές που έχει παραχωρήσει ο χρήστης. Πρωτού πραγματοποιηθεί η αλλαγή των στοιχείων, γίνεται έλεγχος αν τα στοιχεία είναι καταχωρημένα σε άλλον χρήστη.

```

1 <script type="text/javascript">
2     function updateProfile(event) {
3         var http = new XMLHttpRequest();
4         var url = 'update_profile.php';
5         var username = document.getElementById('
profile_username').value;
```

```

6         var name = document.getElementById( 'profile_name' ).value;
7         var surname = document.getElementById( 'profile_surname' ).value;
8         var phone = document.getElementById( 'profile_phone' ).value;
9         var email = document.getElementById( 'profile_email' ).value;
10        var params = 'username=' + username + '&name=' + name
11          + '&surname=' + surname + '&phone=' + phone + '&email=' + email;
12        var link = url + "?" + params;
13
14        http.open( 'GET' , link , false );
15        http.setRequestHeader( 'Content-type' , 'application/x-www-form-urlencoded' );
16        http.onreadystatechange = function() {
17            if( http.readyState == 4 && http.status == 200) {
18                if( http.responseText == "username"){
19                    $('#username_error').modal( 'show' );
20                } else if( http.responseText == "phone"){
21                    $('#phone_error').modal( 'show' );
22                } else if( http.responseText == "email"){
23                    $('#email_error').modal( 'show' );
24                } else {
25                    $('#success').modal( 'show' );
26                }
27            }
28            http.send( null );
29        }
30    </script>

```

Σε κάθε περίπτωση, ο χρήστης ενημερώνεται καταλλήλως με αντίστοιχο μήνυμα για την εξέλιξη των αλλαγών που ζήτησε.

- **update_profile.php**

Το αρχείο δέχεται τις παραμέτρους που στέλνει η σελίδα του **profile.php** και πρώτα ελέγχει αν υπάρχουν δηλωμένα στοιχεία σε άλλον χρήστη. Εφόσον τα νέα στοιχεία δεν υπάρχουν καταχωρημένα, πραγματοποιείται ενημέρωση της βάσης δεδομένων και στη συνέχεια ενημερώνεται και το `$_SESSION` προκειμένου οι αλλαγές να φανούν σε άμεσο χρόνο στο χρήστη.

```

1 <?php
2     require '../settings.php';
3     session_start();
4
5     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
6
7     if (!$conn) {
8         die("Connection failed: " . mysqli_connect_error());
9         echo "failed";
10    }
11    if ($_SERVER['REQUEST_METHOD'] == 'GET')
12    {
13        $username = $_GET['username'];
14        $name = $_GET['name'];
15        $surname = $_GET['surname'];
16        $phone = $_GET['phone'];
17        $email = $_GET['email'];
18        $id = $_SESSION['id'];
19        $checker = true;
20
21        // 0 if strings are the same
22        $username_changes = strcmp($username, $_SESSION['username']);
23        $phone_changes = strcmp($phone, $_SESSION['phone']);
24        $email_changes = strcmp($email, $_SESSION['email']);
25
26        if ($username_changes != 0){
27            // check if username is already in db
28            $sql_username = "SELECT COUNT(username) AS c FROM
users WHERE username = '" . $username . "'";
29            $result = mysqli_query($conn, $sql_username);
30            $check = mysqli_num_rows($result);
31

```

```

32         if( $check > 0){
33             while( $data = mysqli_fetch_assoc($result)){
34                 if( $data[ "c" ] == 1){
35                     // user exist with this username
36                     echo "username";
37                     $checker = false;
38                     break;
39                 }
40             }
41         }
42     }
43
44     if( $phone_changes != 0){
45         // check if phone is already in db
46         $sql_phone = "SELECT COUNT(phone) AS c FROM users
WHERE phone = '" . $phone . "' ";
47         $result = mysqli_query($conn , $sql_phone);
48         $check = mysqli_num_rows($result);
49
50         if( $check > 0){
51             while( $data = mysqli_fetch_assoc($result)){
52                 if( $data[ "c" ] == 1){
53                     // user exist with this phone
54                     echo "phone";
55                     $checker = false;
56                     break;
57                 }
58             }
59         }
60     }
61
62     if( $email_changes != 0){
63         // check if email is already in db
64         $sql_email = "SELECT COUNT(email) AS c FROM users
WHERE email = '" . $email . "' ";
65         $result = mysqli_query($conn , $sql_email);
66         $check = mysqli_num_rows($result);
67
68         if( $check > 0){
69             while( $data = mysqli_fetch_assoc($result)){
70                 if( $data[ "c" ] == 1){

```

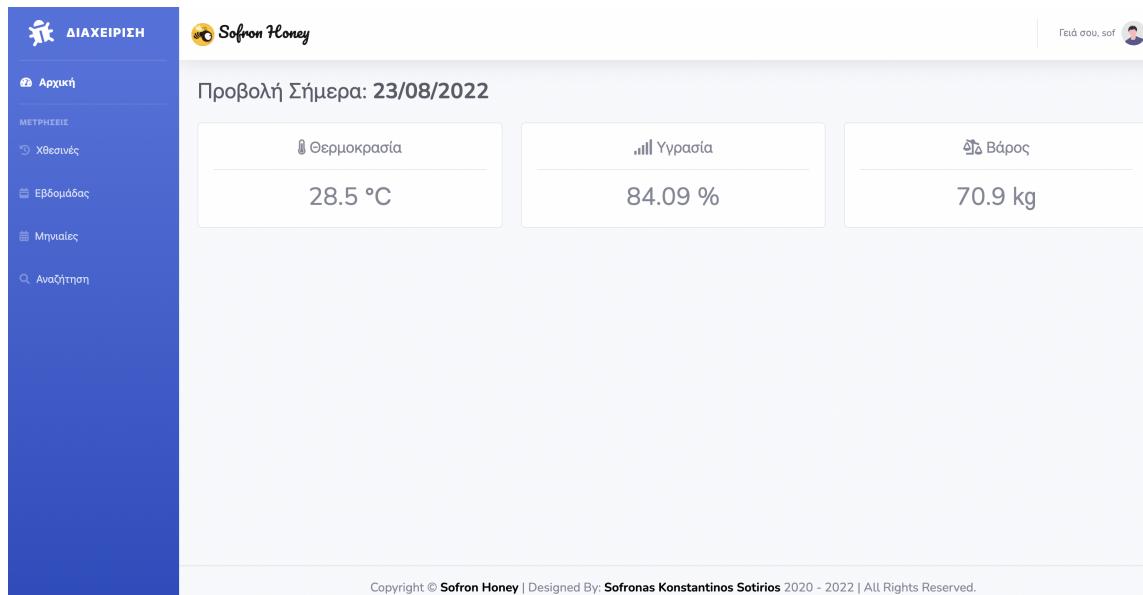
```

71                                     // user exist with this email
72                                     echo "email";
73                                     $checker = false;
74                                     break;
75                                 }
76                             }
77                         }
78                     }
79
80                     if($checker == true){
81                         // no problem
82                         $sql = 'UPDATE users
83                         SET name = "'.$_GET['name'].'
84                         , surname = "'.$_GET['surname'].'
85                         , phone = "'.$_GET['phone'].'
86                         , email = "'.$_GET['email'].'
87                         , username = "'.$_GET['username'].'
88                         ' WHERE id = "'.$_SESSION['id'].'';
89                         // echo $sql .<br>';
90                         $result = mysqli_query($conn, $sql);
91                         if($result == true){
92                             $_SESSION['username'] = $_GET['username'];
93                             $_SESSION['name'] = $_GET['name'];
94                             $_SESSION['surname'] = $_GET['surname'];
95                             $_SESSION['phone'] = $_GET['phone'];
96                             $_SESSION['email'] = $_GET['email'];
97                             $_SESSION['user'] = $_GET['username'];
98                             echo "Changes have been done";
99                         } elseif($result == false){
100                             echo "Cound not make the changes";
101                         }
102                     }
103                 }
104
105                 mysqli_close($conn);
106 ?>
```

5.2.4.9 Περιβάλλον Προβολής δεδομένων

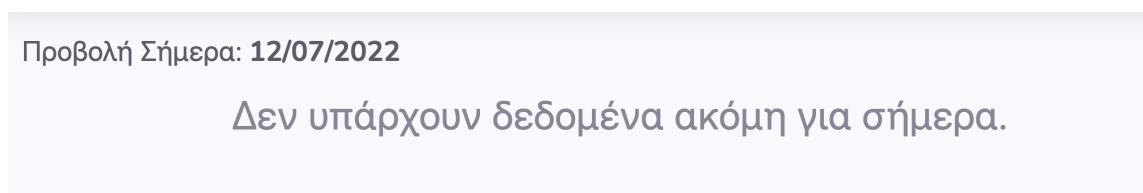
- **management/index.php**

Ο χρήστης πλέον έχει πρόσβαση στο σύστημα, δηλαδή στη παρουσίαση δεδομένων από την ηλεκτρονική ζυγαριά.



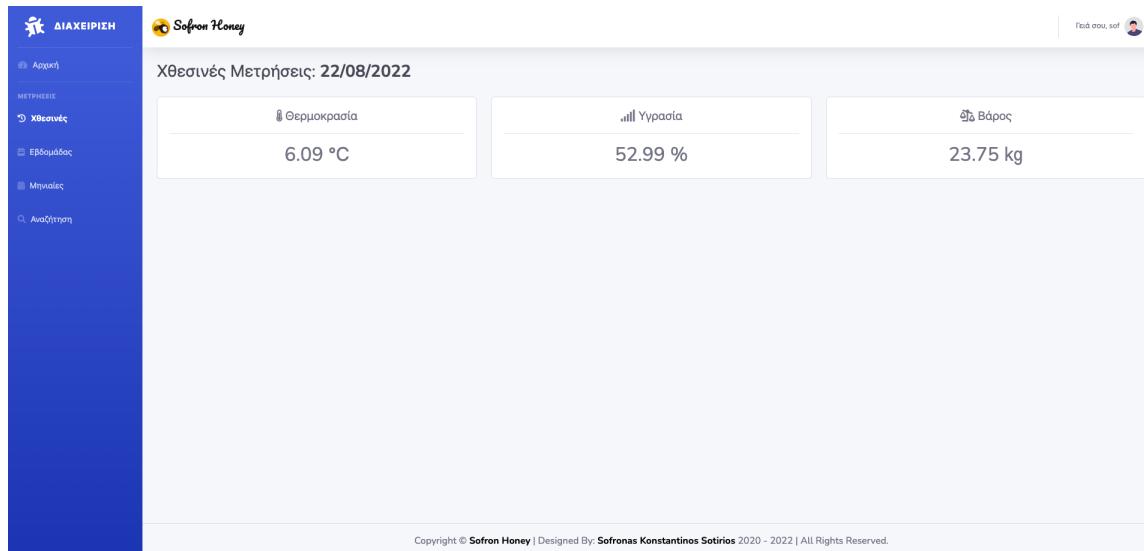
Σχήμα 75: Αρχική Σελίδα Παρουσίασης Δεδομένων

Η σελίδα **index.php**, περιέχει τα στοιχεία της τρέχουσας ημέρας. Σε περίπτωση που η ηλεκτρονική ζυγαριά δεν έχει αποστείλει ακόμη μήνυμα για την ενημέρωση των μετρήσεων, τότε στην θέση των μετρήσεων εμφανίζεται σχετικό μήνυμα.



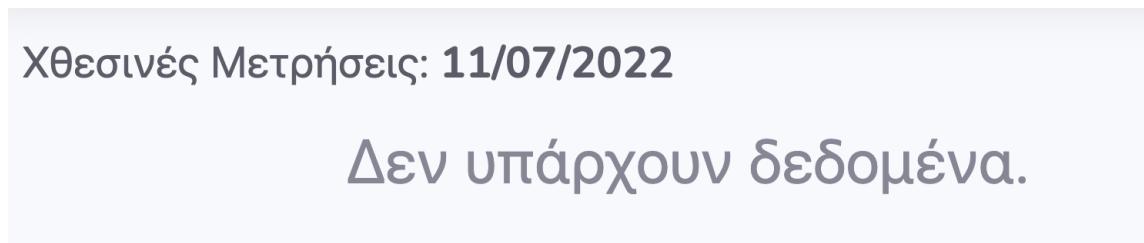
Σχήμα 76: Μήνυμα Μη ύπαρξης Δεδομένων

- **managment/yesterday.php**



Σχήμα 77: Παρουσίαση Δεδομένων Προηγούμενης Ημέρας

Η σελίδα **yesterday.php**, περιέχει τις μετρήσεις της προηγούμενης ημέρας. Σε περίπτωση που η ηλεκτρονική ζυγαριά, για οποιδήποτε λόγο δεν είχε στείλει μήνυμα ή δεν λειτουργούσε την προηγούμενη ημέρα για την ενημέρωση των μετρήσεων, εμφανίζεται σχετικό μήνυμα.



Σχήμα 78: Μήνυμα Μη ύπαρξης Δεδομένων Χθεσινής Ημέρας

- [managment/week.php](#)

The screenshot shows a web-based management system for Sofron Honey. On the left, a blue sidebar contains navigation links: Αρχική, Χθεσινές, Εβδομάδας, Μηνιαίες, and Αναζήτηση. The main content area has a header "Sofron Honey" and a sub-header "Προβολή Εβδομάδας: 26/09/2022 - 02/10/2022". Below this is a table with columns: #, Ημερομηνία, Θερμοκρασία, Υγρασία, and Βάρος. The table rows show data for days 1 through 7 of the week, with values such as 15.01, 89.86, and 94.66. At the bottom of the page, there is a copyright notice: "Copyright © Sofron Honey | Designed By: Sofronas Konstantinos Sotirios 2020 - 2022 | All Rights Reserved." A small profile picture of the developer is also present.

Σχήμα 79: Παρουσίαση Δεδομένων Τρέχουσας Εβδομάδας

Η σελίδα **week.php**, περιέχει τα στοιχεία της τρέχουσας εβδομάδας. Ο τίτλος της σελίδας αναγράφει το χρονικό διάστημα ημερών που περιέχει η τρέχουσα εβδομάδα και υποδεικνύει τη περίοδο μετρήσεων. Για παράδειγμα, αν στην τρέχουσα εβδομάδα η ημέρα είναι Δευτέρα, τότε ο πίνακας δεδομένων θα περιέχει μόνο μιας ημέρας δεδομένα.

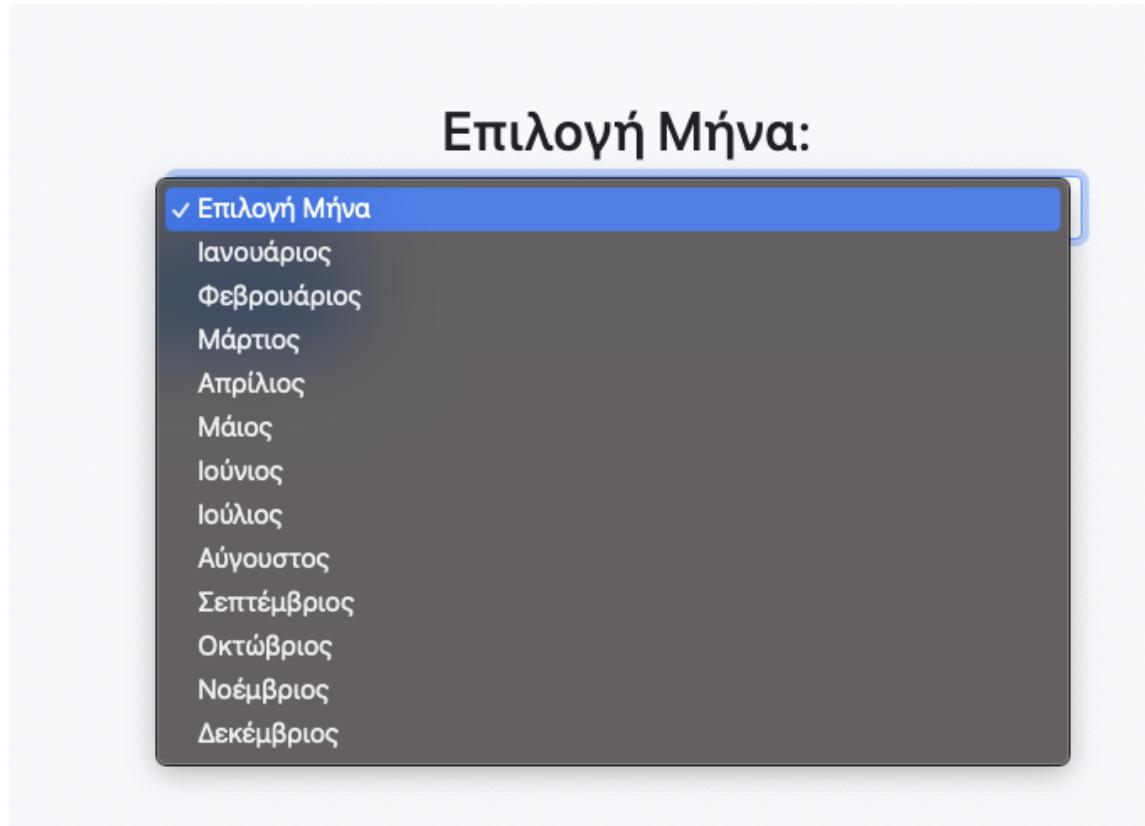
Προβολή Εβδομάδας: 07/07/2022 - 13/07/2022

Δεν υπάρχουν δεδομένα για την τρέχουσα εβδομάδα.

Σχήμα 80: Μήνυμα Μη ύπαρξης Δεδομένων Εβδομάδας

- **managment/month.php**

Η σελίδα **month.php**, περιέχει μια φόρμα αναζήτησης, μέσω αναδύόμενης λίστας επιλογών, για να μπορέσει να επιλέξει ο χρήστης τον επιθυμητό μήνα.

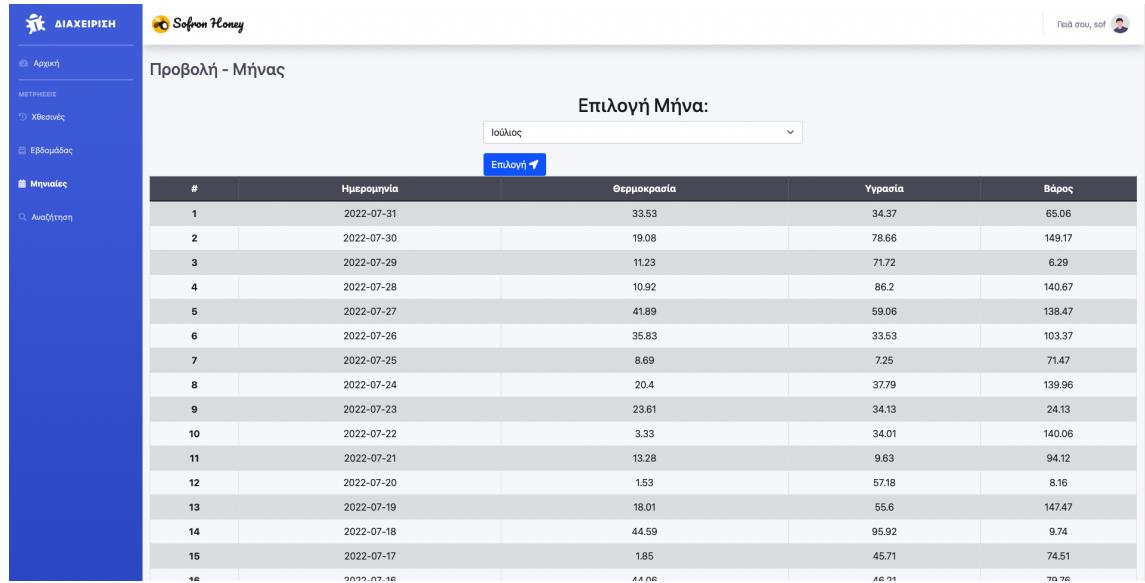


Σχήμα 81: Μενού Επιλογής Μήνα

Ο χρήστης μπορεί να επιλέξει όποιον μήνα επιθυμεί και αφορά το τρέχον έτος. Μόλις επιλέξει μήνα και πατήσει το κουμπί **Επιλογή**, καλείται το αρχείο **getMonthData.php**, μέσα από τη συνάρτηση **caller()** με είσοδο την επιλογή του μήνα σε αριθμό. Η απάντηση που θα στείλει, αποθηκεύεται στο πίνακα **data_table**.

Σε περίπτωση που η απάντηση δεν είναι πίνακας, διότι δεν υπήρξαν δεδομένα για την επιλογή του χρήστη, παρουσιάζεται ένα κείμενο για ενημέρωση.

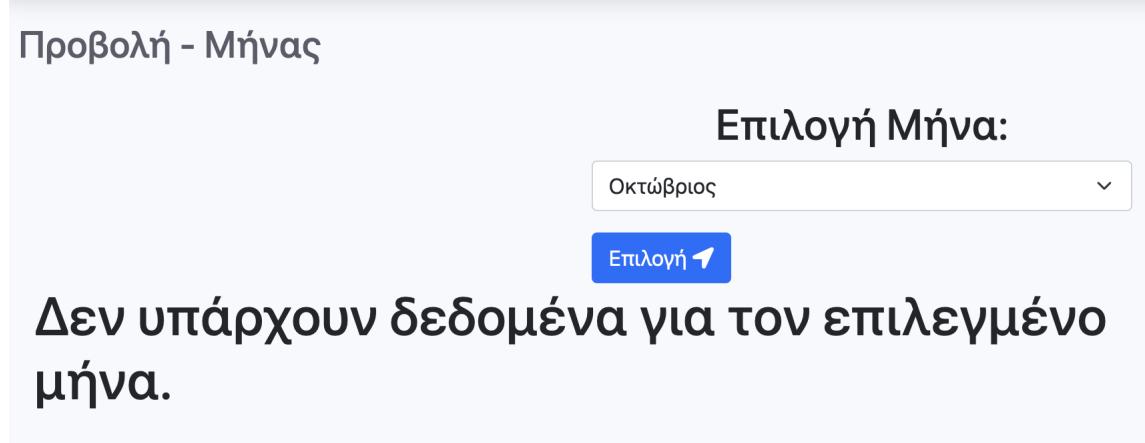
Παρακάτω, παρουσιάζονται τα δεδομένα ενός μήνα που επέλεξε ο χρήστης. Στο συγκεκριμένο παράδειγμα, ο χρήστης επέλεξε το μήνα, Ιούλιο και Οκτώβριο. Ο ένας μήνας, περιέχει δεδομένα ενώ ο άλλος, δεν περιέχει. Παρουσιάζονται οι εμφανίσεις για την αντίστοιχη περίπτωση.



The screenshot shows a software interface for Softron Honey. On the left, there's a sidebar with navigation links: ΔΙΑΧΕΙΡΙΣΗ, Αρχείο, ΜΕΤΡΗΣΕΙΣ, Χθονικός, Εβδομαδιαίς, Μηνιαίες, and Αναζήτηση. The main area is titled "Προβολή - Μήνας" and displays a table titled "Επιλογή Μήνα:". A dropdown menu is open, showing "Ιούλιος". Below it is a blue button labeled "Επιλογή". The table has columns: #, Ημερομηνία, Θερμοκρασία, Υγρασία, and Βάρος. It contains 16 rows of data. At the bottom of the table, there are two buttons: "Επιλογή" and "Έπιπλα".

#	Ημερομηνία	Θερμοκρασία	Υγρασία	Βάρος
1	2022-07-31	33.53	34.37	65.06
2	2022-07-30	19.08	78.66	149.17
3	2022-07-29	11.23	71.72	6.29
4	2022-07-28	10.92	86.2	140.67
5	2022-07-27	41.89	59.06	138.47
6	2022-07-26	35.83	33.53	103.37
7	2022-07-25	8.69	7.25	71.47
8	2022-07-24	20.4	37.79	139.96
9	2022-07-23	23.61	34.13	24.13
10	2022-07-22	3.33	34.01	140.06
11	2022-07-21	13.28	9.63	94.12
12	2022-07-20	1.53	57.18	8.16
13	2022-07-19	18.01	55.6	147.47
14	2022-07-18	44.59	95.92	9.74
15	2022-07-17	1.85	45.71	74.51
16	2022-07-16	8.81	48.71	70.76

Σχήμα 82: Δεδομένα Ιουλίου



The screenshot shows a software interface for Softron Honey. The title is "Προβολή - Μήνας". Below it is a large text area: "Επιλογή Μήνα:" followed by a dropdown menu set to "Οκτώβριος". Below the dropdown is a blue "Επιλογή" button. At the bottom of the text area, there is a large bold text: "Δεν υπάρχουν δεδομένα για τον επιλεγμένο μήνα."

Σχήμα 83: Μήνας χωρίς δεδομένα (Οκτώβριος)

- [managment/getMonthData.php](#)

Το αρχείο καλείται για την αναζήτηση δεδομένων ενός συγκεκριμένου μήνα του τρέχοντος έτους. Εφόσον βρεθούν δεδομένα για το τρέχον μήνα επιστρέφεται ένας πίνακας με τις επικεφαλίδες των πινάκων μαζί και τα δεδομένα.

Η σειρά εμφάνισης των ημερών του μήνα ξεκινάει από τις νεότερες ημέρες καταλήγοντας στις παλαιότερες. Στη περίπτωση που ο μήνας που επέλεξε ο χρήστης, δεν έχει δεδομένα ακόμη, δεν δημιουργείται ο πίνακας και επιστρέφεται ένα κείμενο ενημέρωσης του.

```

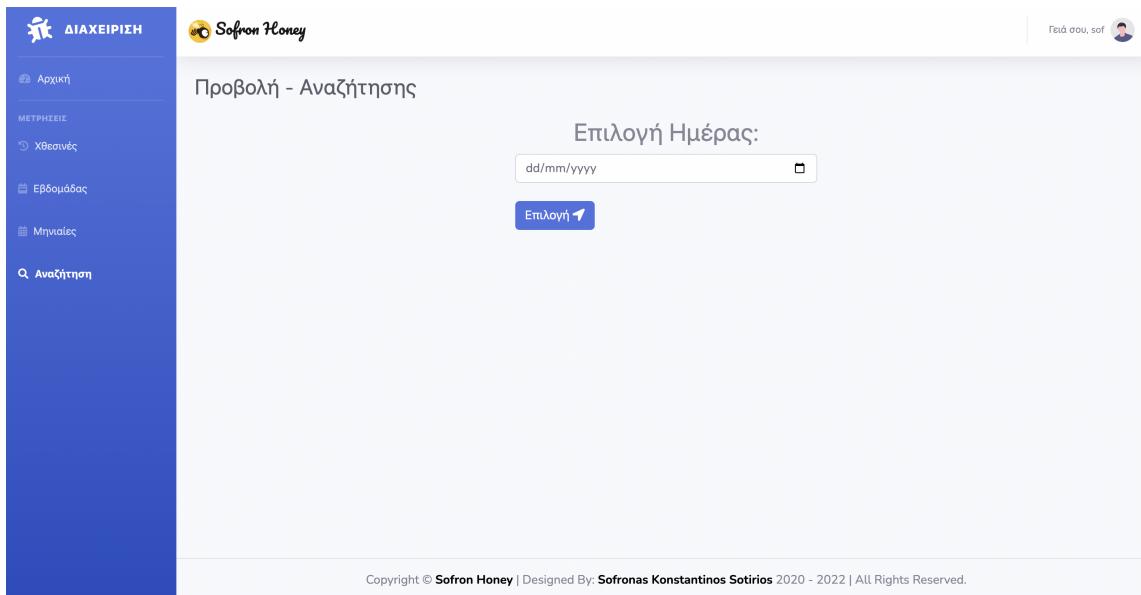
1 <?php
2     require '../settings.php';
3     session_start();
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     if(!$conn){
6         die("Connection failed: " . mysqli_connect_error());
7         echo "failed";
8     }
9     $num = $_POST['m'];
10    $current_year = date("Y");
11    $sql = "SELECT * FROM data
12        WHERE MONTH(imerominia) = ". $num .
13        " AND YEAR(imerominia) =". $current_year . " AND user= ''
14        . $_SESSION['user'] . ''
15        ORDER BY imerominia DESC";
16    $result = mysqli_query($conn, $sql);
17    if (mysqli_num_rows($result) > 0) {
18        echo '<table class="table table-striped table-bordered
19        table-hover" style="text-align: center">
20            <thead class="thead-dark">
21                <tr>

```

```

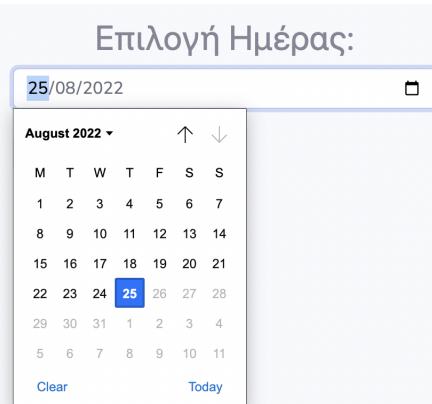
22             <th>Θερμοκρασία</th>
23             <th>Τγρασία</th>
24             <th>Βάρος</th>
25         <tr>
26     </thead>
27     <tbody>
28     ';
29     $counter = 1;
30     while($row = mysqli_fetch_array($result)){
31         echo "<tr>";
32         echo "<th scope='row'>".$counter."</th>";
33         echo "<td>".$row['imerominia']."</td>";
34         echo "<td>".$row['thermokrasia']."</td>";
35         echo "<td>".$row['ugrasia']."</td>";
36         echo "<td>".$row['baros']."</td>";
37         echo "</tr>";
38         $counter = $counter + 1;
39     }
40     echo "</tbody>
41     </table>";
42 } else {
43     echo '
44     <div class="col-md-2"></div>
45     <div class="col-md-8">
46         <h1>Δεν υπάρχουν δεδομένα για τον επιλεγμένο
47         μήνα.</h1>
48     </div>
49     <div class="col-md-2"></div>';
50 ?>
```

- [managment/search.php](#)



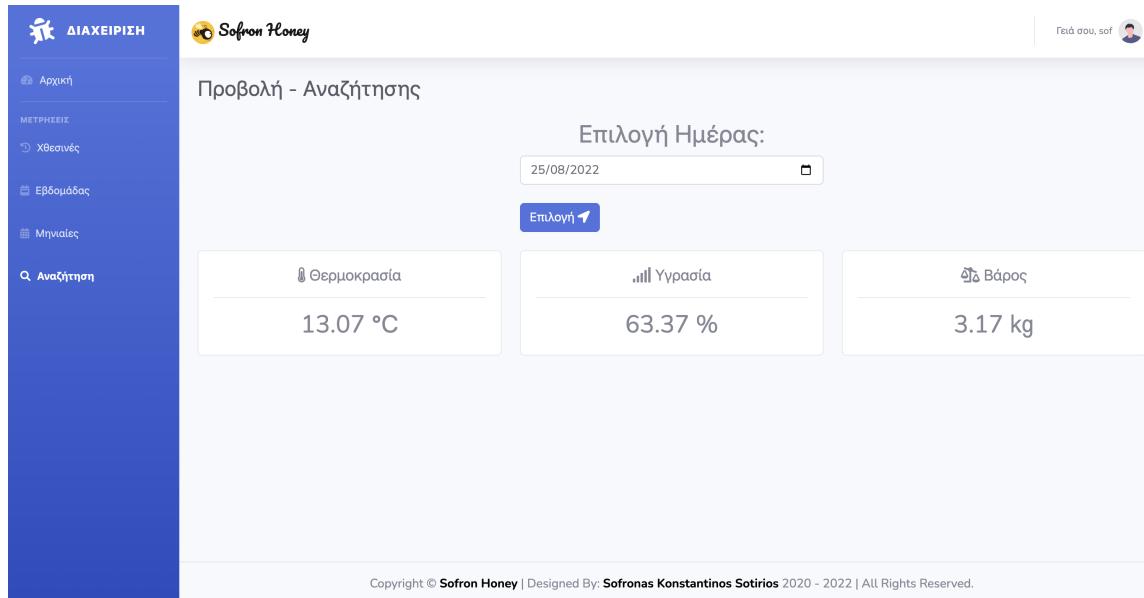
Σχήμα 84: Αναζήτηση Συγκεκριμένης Ημέρας

Ο χρήστης μπορεί να επιλέξει να δει τα δεδομένα συγκεκριμένης ημέρας μέσω της αναζήτησης. Η μέρα μπορεί να είναι οποιαδήποτε ημέρα ανεξάρτητου χρόνου μέχρι την ημέρα πρόσβασης του χρήστη.



Σχήμα 85: Επιλογή Συγκεκριμένης Ημέρας

Τα αποτελέσματα της ημέρας φαίνονται παρακάτω:



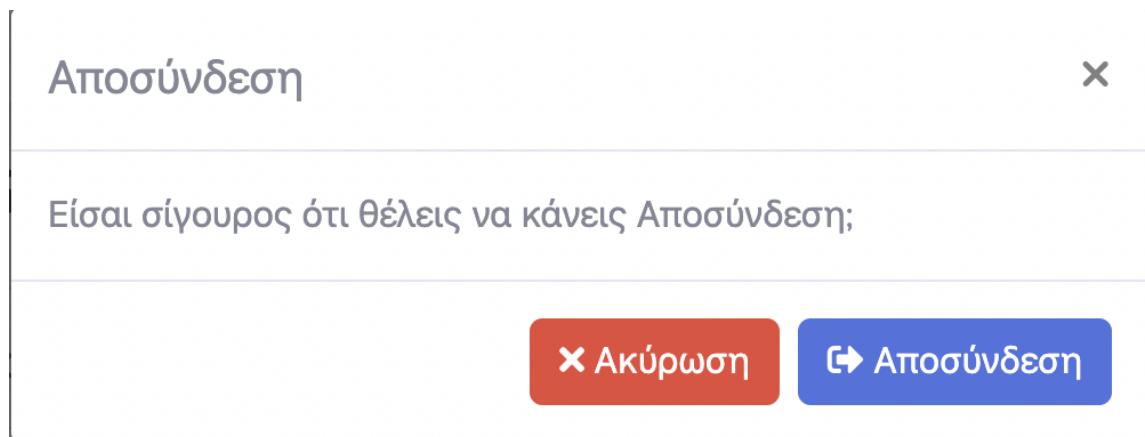
Σχήμα 86: Αποτέλεσμα Συγκεκριμένης Ημέρας

Αυτές είναι όλες οι δυνατότητες που έχει στη κατοχή του ο χρήστης, για τα δεδομένα που γίνονται συλλογή από την ηλεκτρονική ζυγαριά.

5.2.4.10 Αποσύνδεση Χρήστη Από το Σύστημα

- **logout.php**

Όταν ο χρήστης επιθυμεί την αποσύνδεση του από το σύστημα, καλείται το αρχείο **logout.php**.



Σχήμα 87: Αποσύνδεση

```

1 <?php
2     session_start();
3     $_SESSION = array();
4     session_destroy();
5     header("location: index.php");
6     exit;
7 ?>
```

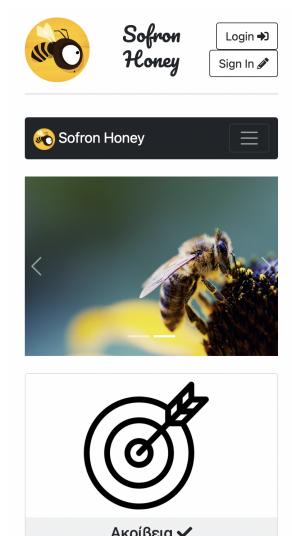
Το αρχείο αποσύνδεσης του χρήστη είναι απλό στην χρήση του. Λαμβάνει όλες τις μεταβλητές που υπάρχουν στο session(στη συνεδρία), τις καταστρέφει και κατευθύνει το χρήστη στην αρχική σελίδα δίχως πλέον να είναι συνδεδεμένος στο σύστημα.

Έχει υλοποιηθεί λειτουργία αυτόματης αποσύνδεσης του χρήστη εντός εξήντα δευτερολέπτων, σε περίπτωση αδράνειας.

```
1 <?php
2     // Initialize the session
3     session_start();
4
5     $inactive = 60; //seconds
6     if (isset($_SESSION['timeout'])) {
7         $session_life = time() - $_SESSION['timeout'];
8         if ($session_life > $inactive)
9             {
10                 session_destroy();
11                 header("Location: Logout.php");
12             }
13     }
14     $_SESSION['timeout'] = time();
15 ?>
```

5.2.4.11 Παρουσίαση Συστήματος από κινητή συσκευή

Η ιστοσελίδα έχει υλοποιηθεί με τέτοιο τρόπο ώστε να είναι συμβατή σε πρόσβαση από κινητές συσκευές. Σε αυτό το μέρος, βοηθάει το framework της, bootstrap, η οποία έχει ήδη φροντίσει να είναι φιλικά προς το χρήστη. Ακολουθούν μερικά στιγμιότυπα:



(α') Αρχική Σελίδα



Πληροφορίες

Τι χρησιμοποιήθηκε στην ηλεκτρονική ζυγαριά;



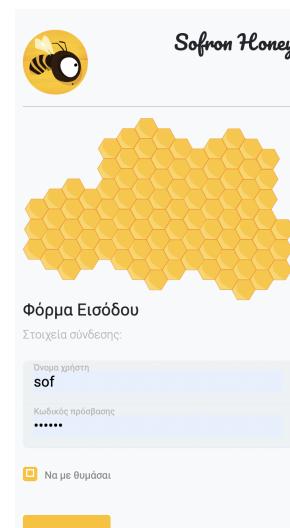
Arduino Uno Rev 3

Ο "εγκέφαλος" της ηλεκτρονικής ζυγαριάς.

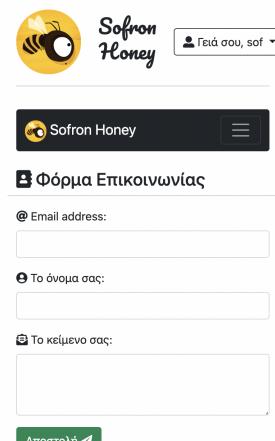


Arduino GSM Shield

(γ') Σελίδα Πληροφοριών



(β') Σελίδα Σύνδεσης



Φόρμα Επικοινωνίας

@ Email address:

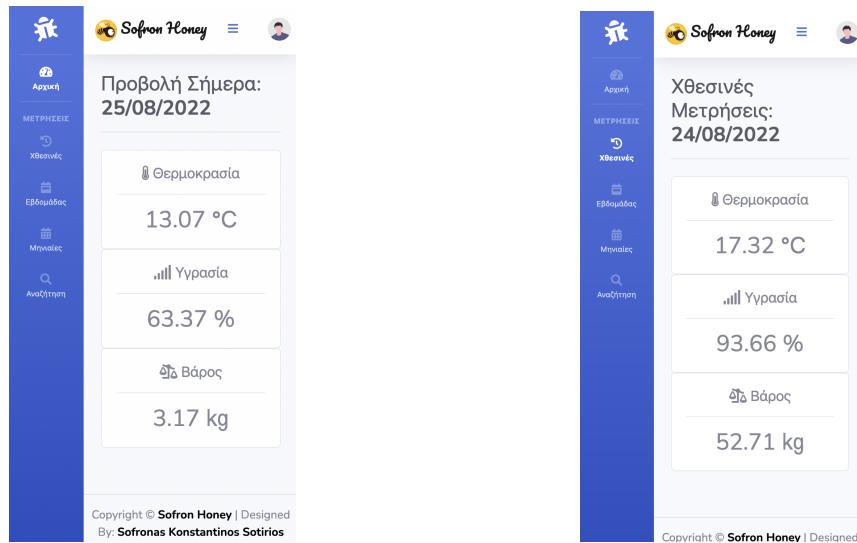
To Όνομα σας:

To κείμενο σας:

Αποστολή

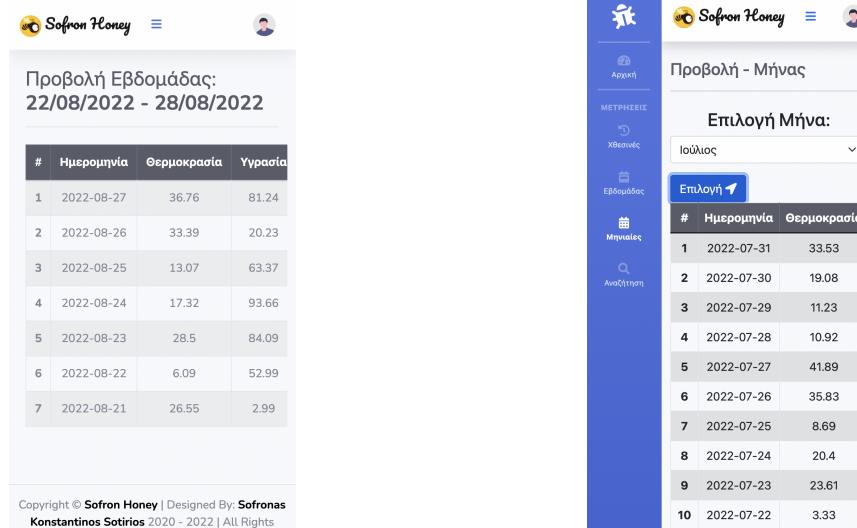
Σχήμα 88: Ιστοσελίδα μέσω Κινητής Συσκευής

Στο διαχειριστικό περιβάλλον, μπορεί να υπάρξει η σύμπτυξη του πλαισίου μενού πατώντας απλά το κουμπί με τις τρεις οριζόντιες γραμμές. Η εικόνα με το μενού σε σύμπτυξη φαίνεται στην εικόνα (γ).



(α') Μετρήσεις Σήμερα

(β') Μετρήσεις του Εχθές



(γ') Μετρήσεις Εβδομάδας

(δ') Μετρήσεις Μήνα

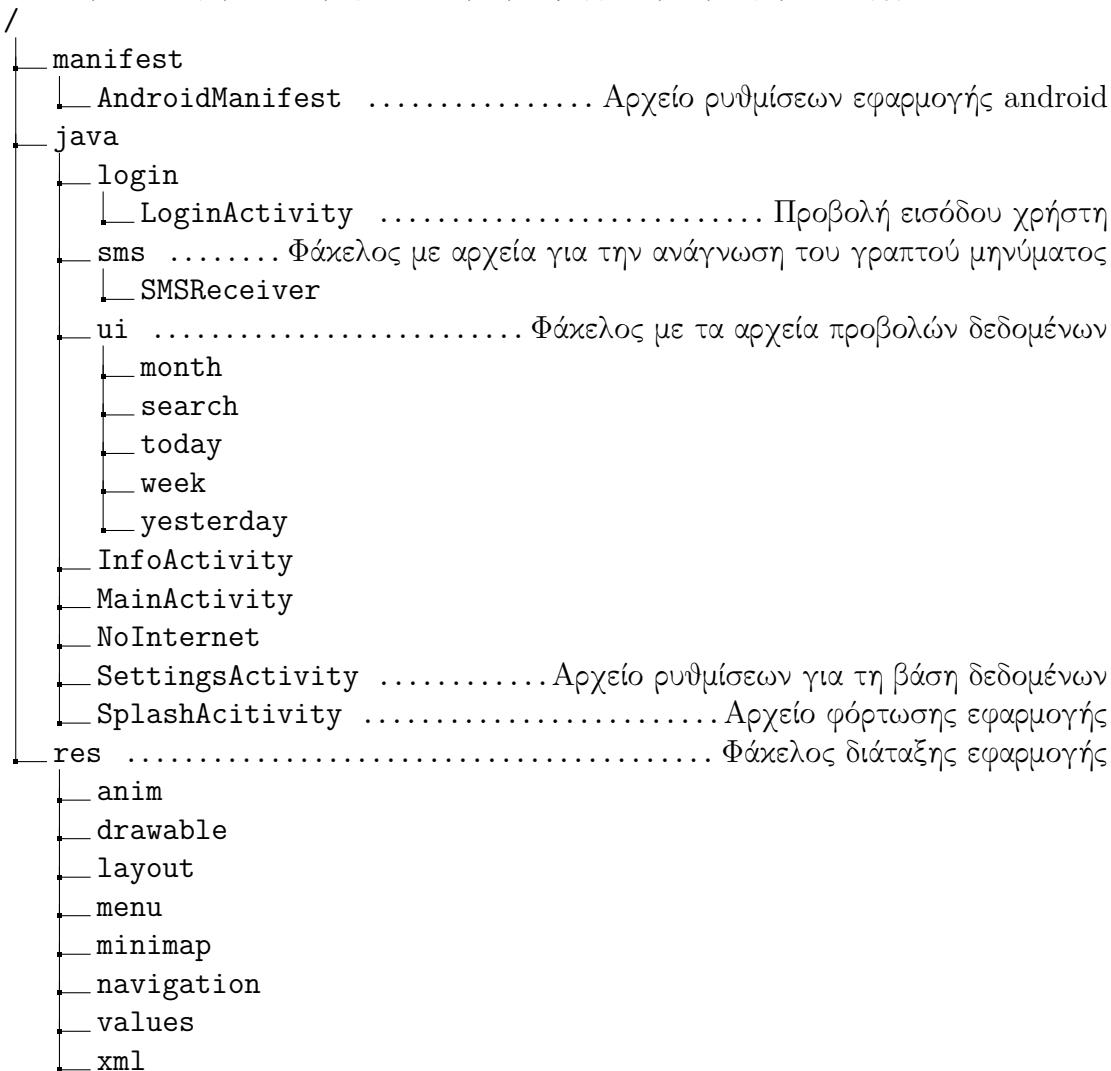
Σχήμα 89: Ιστοσελίδα μέσω Κινητής Συσκευής - Μέρος 2o

5.3 Android - Τηλεματική Εφαρμογή

Για τις ανάγκες ανάπτυξης της τηλεματικής εφαρμογής, χρησιμοποιήθηκε το Android Studio.

5.3.1 Δομή Αρχείων

Για τη καλύτερη δυνατή οργάνωση της εφαρμογής, η δομή των αρχείων:

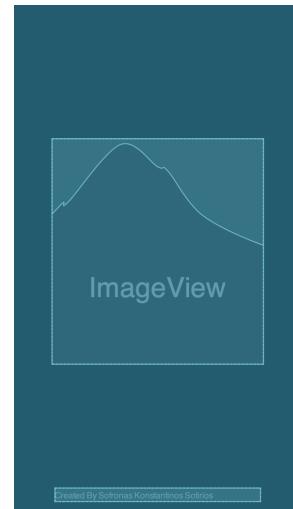


5.3.1.1 Αρχείο προβολής πριν το ξεκίνημα της εφαρμογής

- **SplashActivity.java**



(α') Προβολή εκκίνησης



(β') Προβολή εκκίνησης - Σχεδιασμός xml

Σχήμα 90: Αρχική Προβολή Φόρτωσης Εφαρμογής

```

1 package com.example.sofronas;
2
3 import android.content.Context;
4 import android.content.Intent;
5 import android.net.ConnectivityManager;
6 import android.net.NetworkInfo;
7 import android.os.Bundle;
8 import android.os.Handler;
9 import android.view.animation.Animation;
10 import android.view.animation.AnimationUtils;
11 import android.widget.ImageView;
12
13 import androidx.appcompat.app.AppCompatActivity;
14 import androidx.appcompat.app.AppCompatDelegate;
15
16 import static com.example.sofronas.SettingsActivity.*;

```

```
17 import static com.example.sofronas.SettingsActivity.  
18     getDefaultDark;  
19 import static com.example.sofronas.SettingsActivity.getHost;  
20 import static com.example.sofronas.SettingsActivity.getPassword;  
21 import static com.example.sofronas.SettingsActivity.  
22     getPhoneNumber;  
23 import static com.example.sofronas.SettingsActivity.getUsername;  
24  
25 public class SplashActivity extends AppCompatActivity {  
26     private final int SPLASH_DELAY = 2000;  
27     private ImageView imageView;  
28     private GlobalClass globalVariable;  
29  
30     @Override  
31     protected void onCreate(Bundle savedInstanceState) {  
32         super.onCreate(savedInstanceState);  
33         // Hide title bar for splash screen  
34         try {  
35             this.getSupportActionBar().hide();  
36         } catch (NullPointerException e){  
37             }  
38         setContentView(R.layout.activity_splash);  
39  
40         getWindow().setBackgroundDrawable(null);  
41  
42         // Find image and text to get the animation to work  
43         initializeView();  
44         // Animate logo  
45         animateLogo();  
46  
47         //check if user is connected to the Internet  
48         if(!checkInternet()){  
49             //No internet connection show No internet activity  
50             show404();  
51         } else {  
52             globalVariable = new GlobalClass();  
53             loadSettings();  
54             // Show main activity (Main menu)
```

```

56         goToMain() ;
57     }
58 }
59
60 private void initializeView() {
61     imageView = findViewById(R.id.imageView) ;
62 }
63
64 private void animateLogo() {
65     Animation fadingInAnimation = AnimationUtils .
66     loadAnimation(this ,R.anim.fade_in) ;
67     fadingInAnimation .setDuration(SPLASH_DELAY) ;
68     imageView .startAnimation(fadingInAnimation) ;
69 }
70
71 private void loadSettings(){
72     this .SetupTheme() ;
73     this .SetupPhone() ;
74     this .SetupUser() ;
75     this .SetupPassword() ;
76     this .SetupHost() ;
77 }
78
79 // Setup Theme to startup depends on user choice
80 private void SetupTheme(){
81     Boolean dark_theme = getDefaultDark(this ) ;
82     if (dark_theme == Boolean.TRUE) {
83         AppCompatDelegate.setDefaultNightMode(
84             AppCompatDelegate.MODE_NIGHT_YES) ;
85         globalVariable.setDark_theme(Boolean.TRUE) ;
86     } else {
87         AppCompatDelegate.setDefaultNightMode(
88             AppCompatDelegate.MODE_NIGHT_NO) ;
89         globalVariable.setDark_theme(Boolean.FALSE) ;
90     }
91 }
92
93 // Setup Phone to read SMS depends on user
94 private void SetupPhone(){
95     String phone = getPhoneNumber("phone_number_key" , this ) ;
96     globalVariable.setPhoneNumber(phone) ;

```

```

94 }
95
96 // Setup User to connect Database to read/write data
97 private void SetupUser(){
98     String username = getUsername("username",this);
99     globalVariable.setUsername(username);
100 }
101
102 // Setup Password for user
103 private void SetupPassword(){
104     String password = getPassword("password",this);
105     globalVariable.setPassword(password);
106 }
107
108 // Setup Host URL for connection
109 private void SetupHost(){
110     String host = getHost("url",this);
111     globalVariable.setHost(host);
112 }
113
114 private void show404(){
115     new Handler().postDelayed(()->{
116         startActivity(new Intent(com.example.sofronas.
117             SplashActivity.this, NoInternet.class));
118         finish();
119     }, SPLASH_DELAY);
120 }
121
122 private void goToMain() {
123     new Handler().postDelayed(()->{
124         startActivity(new Intent(com.example.sofronas.
125             SplashActivity.this, MainActivity.class));
126         finish();
127     }, SPLASH_DELAY);
128 }
129
130 private boolean checkInternet(){
131     boolean connected = false;
132     try {
133         ConnectivityManager cm = (ConnectivityManager)
134         getApplicationContext().getSystemService(Context.

```

```

132     CONNECTIVITY_SERVICE) ;
133         NetworkInfo nInfo = cm.getActiveNetworkInfo() ;
134         connected = nInfo != null && nInfo.isAvailable() &&
135         nInfo.isConnected() ;
136         return connected ;
137     } catch (Exception e) {
138         System.out.println(e.getMessage()) ;
139     }
140     return connected ;
}

```

Είναι η πρώτη προβολή, που συναντάει ο χρήστης όταν ξεκινήσει η εφαρμογή. Σκοπός της προβολής αυτής είναι ο έλεγχος συνδεσιμότητας στο Internet με τελικό σκοπό τη πρόσβαση στη βάση δεδομένων.

Εφόσον υπάρχει πρόσβαση στο Internet, γίνεται αρχικοποιήση των ρυθμίσεων του χρήστη για τη πρόσβαση στη βάση δεδομένων και στη συνέχεια η προβολή καλεί την προβολή σύνδεσης - εισόδου.

Σε περίπτωση που η κινητή συσκευή δεν έχει πρόσβαση στο Internet, καλείται η προβολή παρουσίασης της έλειψης Internet . Η εφαρμογή δεν αφήνει να προχωρήσει ο χρήστης περαιτέρω στην εφαμοργή, μέχρι την εξασφάλιση Internet .

5.3.1.2 Αδυναμία Πρόσβασης στο Internet

- **NoInternet.java**



(α') Αδυναμίας σύνδεσης Internet



(β') Αδυναμίας σύνδεσης Internet- Σχεδιασμός xml

Σχήμα 91: Προβολή Αδυναμίας σύνδεσης Internet

Η προβολή που συναντάει ο χρήστης όταν η εφαρμογή, δεν έχει σύνδεση στο Internet με σκοπό την ενημέρωση του. Η έλλειψη Internet δεν επιτρέπει τη σύνδεση με τη βάση δεδομένων.

```

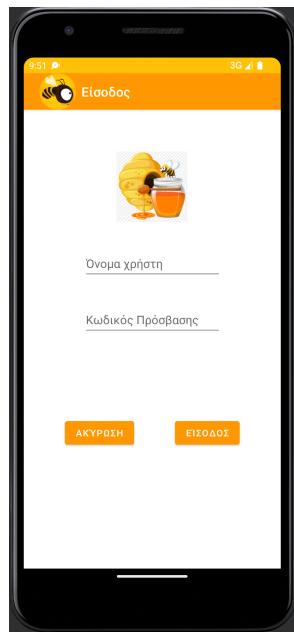
1 package com.example.sofronas;
2
3 import android.os.Bundle;
4 import android.view.View;
5 import android.widget.ImageView;
6
7 import androidx.appcompat.app.ActionBar;
8 import androidx.appcompat.app.AppCompatActivity;
9

```

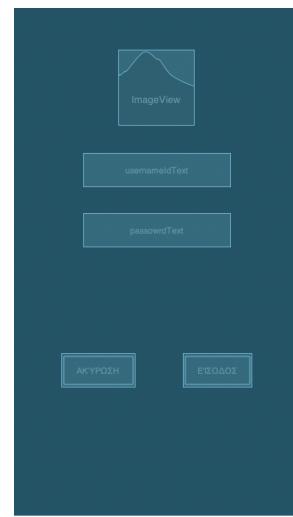
```
10 public class NoInternet extends AppCompatActivity {  
11  
12     @Override  
13     protected void onCreate(Bundle savedInstanceState) {  
14         super.onCreate(savedInstanceState);  
15         setContentView(R.layout.activity_no_internet);  
16  
17         //Set Logo at the top of bar  
18         ActionBar actionBar = getSupportActionBar();  
19         //set up title from a string  
20         actionBar.setTitle(getString(R.string.noInternet));  
21         actionBar.setDisplayHomeAsUpEnabled(true);  
22         actionBar.setIcon(R.mipmap.no_wifi_round);  
23  
24         ImageView closebtn = (ImageView)  
25             ↪ this.findViewById(R.id.imageButton);  
26  
27         closebtn.setOnClickListener(new View.OnClickListener() {  
28             @Override  
29             //close the application  
30             //it cannot work without internet  
31             //no data to display  
32             public void onClick(View v) {  
33                 finish();  
34             }  
35         });  
36     }  
}
```

5.3.1.3 Αρχείο φόρμα σύνδεσης

- **LoginActivity.java**



(a) Προβολή σύνδεσης



(b) Προβολή σύνδεσης - Σχεδιασμός xml

Figure 92: Προβολή φόρμας εισόδου χρήστη

Η προβολή σύνδεσης, θα φορτωθεί στη περίπτωση που ο χρήστης έχει Internet με σκοπό σύνδεσης του με το σύστημα. Ο χρήστης δίνει το όνομα χρήστη (username) που έχει και συμπληρώνει το κωδικό πρόσβασης.

Επιλέγει ‘Είσοδος’. Ξεκινάει ο κατάλληλος έλεγχος με τα δηλωμένα στοιχεία με τη βάση δεδομένων, για τη διαφύλαξη της μέγιστης ασφάλειας. Αν ο ελέγχος ολοκληρωθεί επιτυχώς, τότε προχωράει με την είσοδο στη κεντρική προβολή της εφαρμογής. Στη περίπτωση, λάθος στοιχείων, ο χρήστης ενημερώνεται καταλλήλως και δεν προχωράει στη κεντρική εφαρμογή. Τπάρχει δυνατότητα ακύρωσης εισόδου της εφαρμογής με την επιλογή ‘Ακύρωση’, η οποία τερματίζει την εφαρμογή.

```
1 package com.example.sofronas.login;
2
3 import androidx.appcompat.app.ActionBar;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Context;
7 import android.content.Intent;
8 import android.os.Bundle;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.EditText;
12 import android.widget.Toast;
13
14 import com.android.volley.Request;
15 import com.android.volley.RequestQueue;
16 import com.android.volley.Response;
17 import com.android.volley.VolleyError;
18 import com.android.volley.toolbox.StringRequest;
19 import com.android.volley.toolbox.Volley;
20 import com.example.sofronas.MainActivity;
21 import com.example.sofronas.R;
22
23
24 public class LoginActivity extends AppCompatActivity {
25
26     @Override
27     protected void onCreate(Bundle savedInstanceState) {
28         super.onCreate(savedInstanceState);
29         setContentView(R.layout.activity_login);
30         setTitle(R.string.login_name);
31 }
```

```
32         ActionBar actionBar = getSupportActionBar();  
33  
34         actionBar.setDisplayShowHomeEnabled(true);  
35         actionBar.setIcon(R.mipmap.app_icon);  
36  
37         Button loginBtn = (Button) findViewById(R.id.btnLoginId);  
38         loginBtn.setOnClickListener(new View.OnClickListener() {  
39             @Override  
40             public void onClick(View view) {  
41                 EditText usernameText = (EditText)  
42                     findViewById(R.id.usernameIdText);  
43                 EditText passwordText = (EditText)  
44                     findViewById(R.id.passowrdText);  
45                 checkDataToLogin(usernameText,passwordText);  
46             }  
47         });  
48         Button cancelBtn = (Button) findViewById(R.id.cancelLoginId);  
49         cancelBtn.setOnClickListener(new View.OnClickListener() {  
50             @Override  
51             public void onClick(View view) {  
52                 finishAffinity();  
53                 System.exit(0);  
54             }  
55         });  
56     }  
57     private void checkDataToLogin(EditText username, EditText  
58         password){  
59  
60         String us = username.getText().toString();  
61         String ps = password.getText().toString();  
62  
63         if(us.isEmpty() && ps.isEmpty()){  
64             // Απόδοση στην επόμενη σελίδα  
65             Intent intent = new Intent(getApplicationContext(),  
66                 MainActivity.class);  
67             startActivity(intent);  
68             finish();  
69         }  
70     }  
71 }
```

```

61             System.out.println("Fill data please");
62         }
63         if(us.isEmpty()){
64             System.out.println("Username cannot be empty");
65         }
66         if(ps.isEmpty()) {
67             System.out.println("Password cannot be empty");
68         }
69         if((!ps.isEmpty()) && (!us.isEmpty())){
70             System.out.println("Username: " + us + "\nPassword: " +
71             ps);
72             boolean isReal = checkDB(us,ps);
73         }
74     }
75     private boolean checkDB(String us, String ps){
76         RequestQueue queue = Volley.newRequestQueue(this);
77         String url =
78             "http://192.168.1.4/thesis/check_login.php?username=" + us +
79             "&password=" + ps;
80         // Request a string response from the provided URL.
81         StringRequest stringRequest = new
82             StringRequest(Request.Method.GET, url,
83                 new Response.Listener<String>() {
84                     @Override
85                     public void onResponse(String response) {
86                         // Display the first 500 characters of the
87                         // response string.
88                         if(response.equals("1")){
89                             Intent intent = new
90                             Intent(LoginActivity.this, MainActivity.class);
91                             startActivity(intent);
92                         }
93                     }
94                 });
95         queue.add(stringRequest);
96     }

```

```
87                     finish();
88     } else{
89         Context context =
90             getApplicationContext();
91         CharSequence text = "Λάθος Δεδομένα";
92         int duration = Toast.LENGTH_LONG;
93
94         Toast toast = Toast.makeText(context,
95             text, duration);
96         toast.show();
97         System.out.println("toast shown()");
98     }
99 }
100     }, new Response.ErrorListener() {
101     @Override
102     public void onErrorResponse(VolleyError error) {
103         System.out.println("Error: "+ error);
104     }
105 });
106     queue.add(stringRequest);
107     return false;
108 }
109 }
```

5.3.1.4 Αρχείο κεντρικής προβολής

- >MainActivity.java

Η κύρια προβολή της εφαρμογής η οποία αποτελεί μια κενή προβολή. Στην αρχή, γίνεται ορισμός της πάνω μεριάς της εφαρμογής. Προχωράει η φόρτωση του μενού στη κάτω μεριά, το οποίο περιέχει, τις μετρήσεις για τη τρέχουσα ημέρα, τη χθεσινή, της εβδομάδας, την αναζήτηση μετρήσεων συγκεκριμένου μήνα του τρέχοντος έτους και τέλος, της αναζήτησης συγκεκριμένης ημέρας. Ακόμη, στην επάνω μεριά υπάρχει μενού με λίστα επιλογών για την επιλογή πληροφοριών, ρυθμίσεων αλλά και η επιλογή της αποσύνδεσης του χρήστη από τη τηλεματική εφαρμογή.

Το αρχείο των διατάξεων των μενού βρίσκεται στο φάκελο layouts.

```

1 package com.example.sofronas;
2
3 import android.content.Context;
4 import android.content.Intent;
5 import android.net.ConnectivityManager;
6 import android.net.NetworkInfo;
7 import android.os.Bundle;
8 import android.view.Menu;
9 import android.view.MenuInflater;
10 import android.view.MenuItem;
11 import android.widget.Toast;
12
13 import com.example.sofronas.login.LoginActivity;
14 import com.example.sofronas.sms.SmsReceiver;
15 import com.google.android.material.bottomnavigation.
    BottomNavigationView;
16
17 import androidx.appcompat.app.ActionBar;
18 import androidx.appcompat.app.AppCompatActivity;
19 import androidx.navigation.NavController;
20 import androidx.navigation.Navigation;
21 import androidx.navigation.ui.AppBarConfiguration;
22 import androidx.navigation.ui.NavigationUI;
23
24 public class MainActivity extends AppCompatActivity {
25
26     @Override
27     protected void onCreate(Bundle savedInstanceState) {

```

```

28     super.onCreate(savedInstanceState);
29     setContentView(R.layout.activity_main);

30
31     //Add logo to the bar at left corner
32     ActionBar actionBar = getActionBar();
33     actionBar.setDisplayHomeAsUpEnabled(true);
34     actionBar.setIcon(R.mipmap.bees_foreground_round);

35
36     if (isConnected()) {
37     } else {
38         Toast.makeText(getApplicationContext(), "No Internet
39         Connection", Toast.LENGTH_SHORT).show();
40         Intent myIntent1 = new Intent(MainActivity.this,
41         NoInternet.class);
42         MainActivity.this.startActivity(myIntent1);
43         //close app
44         finish();
45     }
46     BottomNavigationView navView = findViewById(R.id.nav_view);
47
48     // Passing each menu ID as a set of IDs because each
49     // menu should be considered as top level destinations.
50     AppBarConfiguration appBarConfiguration = new
51     AppBarConfiguration.Builder(
52         R.id.navigation_today, R.id.navigation_yesterday,
53         R.id.navigation_week, R.id.navigation_month, R.id.
54         navigation_search)
55         .build();
56     NavController navController = Navigation.
57     findNavController(this, R.id.nav_host_fragment);
58     NavigationUI.setupActionBarWithNavController(this,
59     navController, appBarConfiguration);
60     NavigationUI.setupWithNavController(navView,
61     navController);
62 }

63
64     //function that check if the user is connected to Internet
65     public boolean isConnected() {
66         boolean connected = false;
67         try {
68             ConnectivityManager cm = (ConnectivityManager)

```

```

getApplicationContext() . getSystemService ( Context .
CONNECTIVITY_SERVICE ) ;
    NetworkInfo nInfo = cm . getActiveNetworkInfo () ;
    connected = nInfo != null && nInfo . isAvailable () &&
nInfo . isConnected () ;
    return connected ;
} catch ( Exception e ) {
    System . out . println ( e . getMessage () );
}
return connected ;
}

//Add bottom navigation menu at the bottom
@Override
public boolean onCreateOptionsMenu ( Menu menu ) {
    MenuInflater inflater = getMenuInflater () ;
    inflater . inflate ( R . menu . settingsmenu , menu ) ;
    return super . onCreateOptionsMenu ( menu ) ;
}

//The menu at the right side of application
//Settings and Information and Logout
@Override
public boolean onOptionsItemSelected ( MenuItem item ) {
    switch ( item . getItemId () ) {
        case R . id . action_settings :
            // Settings Page
            Intent myIntent = new Intent ( MainActivity . this ,
SettingsActivity . class ) ;
            MainActivity . this . startActivity ( myIntent ) ;
            return true ;
        case R . id . info_app :
            // Information Page
            Intent info = new Intent ( MainActivity . this ,
InfoActivity . class ) ;
            MainActivity . this . startActivity ( info ) ;
            return true ;
        case R . id . logout_app :
            //Logout User
            Intent logt = new Intent ( MainActivity . this ,
LoginActivity . class ) ;
    }
}

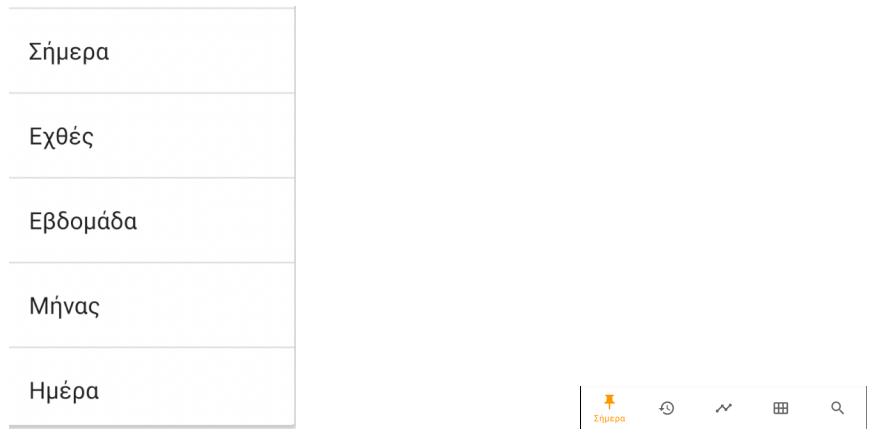
```

```

95         MainActivity.this.startActivity(logt);
96         finish();
97         return false;
98     default:
99         return super.onOptionsItemSelected(item);
100    }
101}
102}

```

Ο χρήστης μπορεί να πλοηγηθεί από την κάτω μπάρα, η οποία περιέχει μενού επιλογών. (Σήμερα/Εχθές/Εβδομάδα/Μήνας/Ημέρα)



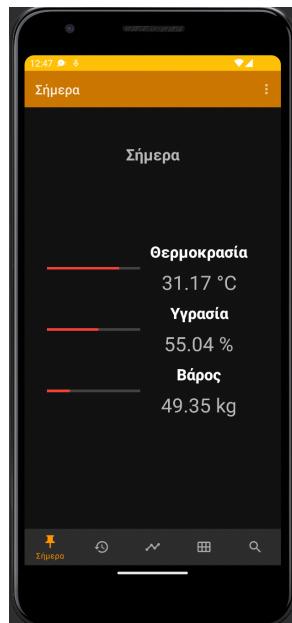
(α') Σχεδιασμός Μενού Κάτω

(β') Εμφάνιση - Μενού Κάτω

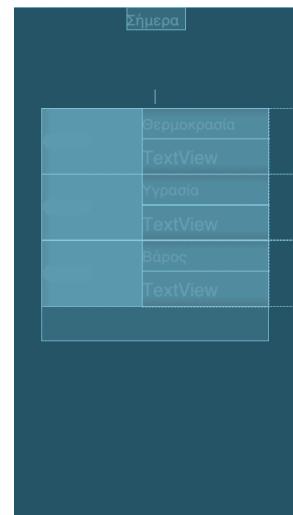
Σχήμα 93: Μενού Επιλογών Κάτω

5.3.1.5 Προβολή: Σήμερα

- **FragmentToday.java**



(α') Προβολή μετρήσεων: Σήμερα



(β') Προβολή μετρήσεων: Σήμερα - Σχεδιασμός xml

Σχήμα 94: Προβολή Δεδομένων - Σήμερα

Πρόκειται για το τμήμα (fragment), το οποίο από προεπιλογή εμφανίζεται πρώτο στη κεντρική προβολή του αρχείου MainActivity. Ενημερώνει το χρήστη για τη μέτρηση που έχει λάβει η ηλεκτρονική ζυγαριά για τη τρέχουσα ημέρα. Οι μετρήσεις αφορούν τη θερμοκρασία, την υγρασία και το βάρος της κυψέλης. Οι μετρήσεις αναγράφονται τόσο σε αριθμούς, για να είναι άμεσα κατανοητοί από το χρήστη, όσο και σε γραμμή προόδου (progress bar). Ο σκοπός ύπαρξης δύο τρόπων αναπαράστασης δεδομένων, είναι για την καλύτερη οπτικοποίηση των μετρήσεων.

```

1 package com.example.sofronas.ui.today;
2 import android.content.SharedPreferences;
3 import android.os.Bundle;
4 import android.view.LayoutInflater;
5 import android.view.View;
```

```
6 import android.view.ViewGroup;
7 import android.widget.ProgressBar;
8 import android.widget.TextView;
9 import android.widget.Toast;
10 import androidx.annotation.NonNull;
11 import androidx.fragment.app.Fragment;
12 import androidx.preference.PreferenceManager;
13 import com.android.volley.Request;
14 import com.android.volley.RequestQueue;
15 import com.android.volley.Response;
16 import com.android.volley.VolleyError;
17 import com.android.volley.toolbox.StringRequest;
18 import com.android.volley.toolbox.Volley;
19 import com.example.sofronas.GlobalClass;
20 import com.example.sofronas.R;
21
22 public class FragmentToday extends Fragment {
23
24     public View onCreateView(@NonNull LayoutInflater inflater,
25                             ViewGroup container, Bundle
26                             savedInstanceState) {
27         View root = inflater.inflate(R.layout.fragment_today,
28                             container, false);
29         s(root);
30         GlobalClass s = new GlobalClass();
31
32         // Instantiate the RequestQueue.
33         RequestQueue queue = Volley.newRequestQueue(getActivity());
34         SharedPreferences settings =
35         PreferenceManager.getDefaultSharedPreferences(getActivity());
36         String ur = settings.getString("url", s.getHost());
37         String us = settings.getString("username_id", "sof");
38         String url = "http://" + ur +
39         "/thesis/android/getToday.php?us=" + us;
40
41         System.out.println(url);
42
43         // Request a string response from the provided URL.
```

```

40     StringRequest stringRequest = new
41     ↪ StringRequest(Request.Method.GET, url,
42         new Response.Listener<String>() {
43             @Override
44             public void onResponse(String response) {
45                 // Display the first 500 characters of the
46                 ↪ response string.
47                 //System.out.println("Response is: "+
48                 ↪ response);
49
50             /* Scenario
51             empty string = no data today
52             data:
53                 first = temp
54                 second = hum
55                 third = weight
56             Analysis:
57                 check if data are float
58                 to fill bar with its int value
59             */
60             Boolean check = false;
61             if(response.equals("")) {
62                 check = true;
63             }
64             if(check == false){
65                 String[] data =
66                 ↪ response.toString().split("<br>");
67                 TextView temp =
68                 ↪ (TextView)root.findViewById(R.id.texttemp);
69                 ProgressBar tempBar =
70                 ↪ (ProgressBar)root.findViewById(R.id.progressBart);
71                 try {
72                     Float te = Float.parseFloat(data[0]);
73                     Integer k = te.intValue();
74                     tempBar.setProgress(k);
75                 } catch (NumberFormatException e) {
76
77                 ↪ tempBar.setProgress(Integer.parseInt(data[0]));
78             }
79         }
80     }
81 }
```

```
71                         }
72                         temp.setText(data[0] + " " + (char)
73                         ↪ 0x00B0+"C"));
74
75                         TextView hum =
76                         ↪ (TextView)root.findViewById(R.id.text_hum);
77                         ProgressBar humBar =
78                         ↪ (ProgressBar)root.findViewById(R.id.progressBarh);
79                         try {
80                             Float hu = Float.parseFloat(data[1]);
81                             Integer h = hu.intValue();
82                             humBar.setProgress(h);
83                         } catch (NumberFormatException e){
84
85                         ↪ humBar.setProgress(Integer.parseInt(data[1]));
86                         ↪
87                         hum.setText(data[1] + " %");
88
89                         TextView wei =
90                         ↪ (TextView)root.findViewById(R.id.text_wei);
91                         ProgressBar weiBar =
92                         ↪ (ProgressBar)root.findViewById(R.id.progressBarw);
93                         try{
94                            Float tem =
95                             ↪ Float.parseFloat(data[2]);
96                             Integer i = tem.intValue();
97                             weiBar.setProgress(i);
98                         } catch (NumberFormatException e) {
99
100                         ↪ weiBar.setProgress(Integer.parseInt(data[2]));
101                         ↪
102                         wei.setText(data[2] + " kg");
103                     } else {
104                         TextView temp =
105                         ↪ (TextView)root.findViewById(R.id.texttemp);
106                         ProgressBar tempBar =
107                         ↪ (ProgressBar)root.findViewById(R.id.progressBart);
108                         tempBar.setProgress(0);
```

```

99                     temp.setText("–");
100
101                 TextView hum =
102             ↵ (TextView)root.findViewById(R.id.text_hum);
103                 ProgressBar humBar =
104             ↵ (ProgressBar)root.findViewById(R.id.progressBarH);
105                     humBar.setProgress(0);
106                     hum.setText("–");
107
108                 TextView wei =
109             ↵ (TextView)root.findViewById(R.id.text_wei);
110                 ProgressBar weiBar =
111             ↵ (ProgressBar)root.findViewById(R.id.progressBarW);
112                     weiBar.setProgress(0);
113                     wei.setText("–");
114             }
115         }
116     }, new Response.ErrorListener() {
117         @Override
118         public void onErrorResponse(VolleyError error) {
119             System.out.println("Error: "+ error);
120             System.out.println("That didn't work!");
121             TextView temp =
122             ↵ (TextView)root.findViewById(R.id.texttemp);
123             ProgressBar tempBar =
124             ↵ (ProgressBar)root.findViewById(R.id.progressBarT);
125                     tempBar.setProgress(0);
126                     temp.setText("–");
127
128                 TextView hum =
129             ↵ (TextView)root.findViewById(R.id.text_hum);
130                 ProgressBar humBar =
131             ↵ (ProgressBar)root.findViewById(R.id.progressBarH);
132                     humBar.setProgress(0);
133                     hum.setText("–");
134
135                 TextView wei =
136             ↵ (TextView)root.findViewById(R.id.text_wei);

```

```

128         ProgressBar weiBar =
129             ↪ (ProgressBar)root.findViewById(R.id.progressBarw);
130             weiBar.setProgress(0);
131             weiBar.setText("-");

132             Toast t = Toast.makeText(root.getContext(),
133             ↪ error.toString(), Toast.LENGTH_SHORT);
134             t.show();
135         });
136         // Add the request to the RequestQueue.
137         queue.add(stringRequest);
138         return root;
139     }

140
141     private void s(View root){
142         TextView temp = (TextView)root.findViewById(R.id.texttemp);
143         ProgressBar tempBar =
144             ↪ (ProgressBar)root.findViewById(R.id.progressBart);
145             tempBar.setProgress(0);
146             tempBar.setText("-");

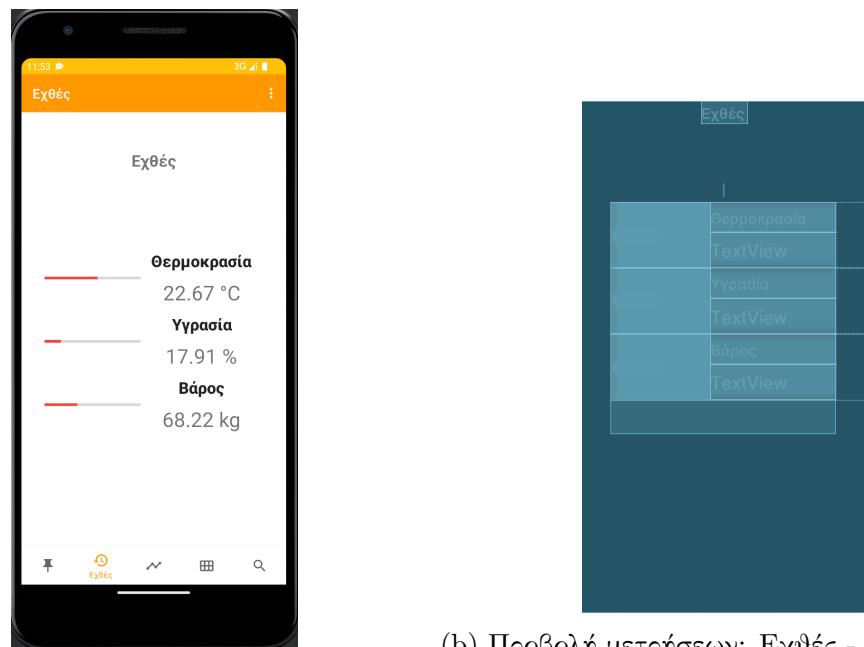
147         TextView hum = (TextView)root.findViewById(R.id.text_hum);
148         ProgressBar humBar =
149             ↪ (ProgressBar)root.findViewById(R.id.progressBarh);
150             humBar.setProgress(0);
151             humBar.setText("-");

152         TextView wei = (TextView)root.findViewById(R.id.text_wei);
153         ProgressBar weiBar =
154             ↪ (ProgressBar)root.findViewById(R.id.progressBarw);
155             weiBar.setProgress(0);
156             weiBar.setText("-");
157     }
}

```

5.3.1.6 Προβολή: Εχθές

- FragmentYesterday.java



(a) Προβολή μετρήσεων: Εχθές

(b) Προβολή μετρήσεων: Εχθές - Σχεδιασμός xml

Figure 95: Προβολή Δεδομένων - Εχθές

Πρόκειται για τη δεύτερη προβολή μετρήσεων. Οι μετρήσεις αφορούν τη χθεσινή ημέρα. Ο λόγος ύπαρξης αυτής της προβολής είναι η άμεση σύγχριση τιμών μεταξύ της σημερινής και της χθεσινής ημέρας, από το χρήστη.

```

1 package com.example.sofronas.ui.yesterday;
2
3 import android.content.SharedPreferences;
4 import android.os.Bundle;
5 import android.view.LayoutInflater;
6 import android.view.View;
7 import android.view.ViewGroup;
8 import android.widget.ProgressBar;
9 import android.widget.TextView;
```

```
10 import android.widget.Toast;
11
12 import androidx.annotation.NonNull;
13 import androidx.fragment.app.Fragment;
14 import androidx.lifecycle.ViewModelProvider;
15 import androidx.preference.PreferenceManager;
16
17 import com.android.volley.Request;
18 import com.android.volley.RequestQueue;
19 import com.android.volley.Response;
20 import com.android.volley.VolleyError;
21 import com.android.volley.toolbox.StringRequest;
22 import com.android.volley.toolbox.Volley;
23 import com.example.sofronas.R;
24
25 public class FragmentYesterday extends Fragment {
26
27     public View onCreateView(@NonNull LayoutInflater inflater,
28                             ViewGroup container, Bundle
29     ↪ savedInstanceState) {
30         View root = inflater.inflate(R.layout.fragment_yesterday,
31     ↪ container, false);
32         s(root);
33
34         // Instantiate the RequestQueue.
35         RequestQueue queue = Volley.newRequestQueue(getActivity());
36         Sharedpreferences settings =
37     ↪ PreferenceManager.getDefaultSharedPreferences(getActivity());
38         String ur = settings.getString("url", "127.0.0.1");
39         String us = settings.getString("username_id", "sof");
40         String url = "http://" + ur +
41     ↪ "/thesis/android/getYesterday.php?us=" +us;
42
43         // Request a string response from the provided URL.
44         StringRequest stringRequest = new
45     ↪ StringRequest(Request.Method.POST, url,
46             new Response.Listener<String>() {
47                 @Override
```

```

43             public void onResponse(String response) {
44                 // Display the first 500 characters of the
45                 ↵ response string.
46                 //System.out.println("Response is: "+
47                 ↵ response);
48                 Boolean check = false;
49                 if(response.equals("")) {
50                     check = true;
51                 }
52                 if(check == false){
53                     String[] data =
54                     ↵ response.toString().split("<br>");
55                     TextView temp =
56                     ↵ (TextView)root.findViewById(R.id.texttemp_y);
57                     ProgressBar tempBar =
58                     ↵ (ProgressBar)root.findViewById(R.id.progressBary);
59                     try {
60                         Float te = Float.parseFloat(data[0]);
61                         Integer k = te.intValue();
62                         tempBar.setProgress(k);
63                     } catch (NumberFormatException e) {
64
65                         ↵ tempBar.setProgress(Integer.parseInt(data[0]));
66                         }
67                         temp.setText(data[0] + " " + (char)
68                         ↵ 0x00B0+"C");
69
70                         TextView hum =
71                         ↵ (TextView)root.findViewById(R.id.text_hum_y);
72                         ProgressBar humBar =
73                         ↵ (ProgressBar)root.findViewById(R.id.progressBarhy);
74                         try {
75                             Float hu = Float.parseFloat(data[1]);
76                             Integer h = hu.intValue();
77                             humBar.setProgress(h);
78                         } catch (NumberFormatException e){
79
80                         ↵ humBar.setProgress(Integer.parseInt(data[1]));

```

```

71                         }
72                         hum.setText(data[1] + " %");
73
74                     TextView wei =
75                     ↳ (TextView)root.findViewById(R.id.text_wei_y);
76                     ProgressBar weiBar =
77                     ↳ (ProgressBar)root.findViewById(R.id.progressBarw_y);
78                     try{
79                         Float tem =
80                         ↳ Float.parseFloat(data[2]);
81                         Integer i = tem.intValue();
82                         weiBar.setProgress(i);
83                     } catch (NumberFormatException e) {
84
85                         ↳ weiBar.setProgress(Integer.parseInt(data[2]));
86                         }
87                         wei.setText(data[2] + " kg");
88                     } else {
89                         TextView temp =
90                         ↳ (TextView)root.findViewById(R.id.texttemp_y);
91                         ProgressBar tempBar =
92                         ↳ (ProgressBar)root.findViewById(R.id.progressBarBary);
93                         tempBar.setProgress(0);
94                         temp.setText("—");
95
96                         TextView hum =
97                         ↳ (TextView)root.findViewById(R.id.text_hum_y);
98                         ProgressBar humBar =
99                         ↳ (ProgressBar)root.findViewById(R.id.progressBarBarhy);
100                         humBar.setProgress(0);
101                         hum.setText("—");
102
103                         TextView wei =
104                         ↳ (TextView)root.findViewById(R.id.text_wei_y);
105                         ProgressBar weiBar =
106                         ↳ (ProgressBar)root.findViewById(R.id.progressBarw_y);
107                         weiBar.setProgress(0);
108                         wei.setText("—");

```

```

99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
    }
}
}, new Response.ErrorListener() {
@Override
public void onErrorResponse(VolleyError error) {
    System.out.println("Error: " + error);
    System.out.println("That didn't work!");

    TextView temp =
    ↪ (TextView)root.findViewById(R.id.texttemp_y);
    ProgressBar tempBar =
    ↪ (ProgressBar)root.findViewById(R.id.progressBary);
    tempBar.setProgress(0);
    temp.setText("-");

    TextView hum =
    ↪ (TextView)root.findViewById(R.id.text_hum_y);
    ProgressBar humBar =
    ↪ (ProgressBar)root.findViewById(R.id.progressBarhy);
    humBar.setProgress(0);
    hum.setText("-");

    TextView wei =
    ↪ (TextView)root.findViewById(R.id.text_wei_y);
    ProgressBar weiBar =
    ↪ (ProgressBar)root.findViewById(R.id.progressBarw_y);
    weiBar.setProgress(0);
    wei.setText("-");

    Toast t = Toast.makeText(root.getContext(),
    ↪ error.toString(), Toast.LENGTH_SHORT);
    t.show();
}
});

// Add the request to the RequestQueue.
queue.add(stringRequest);

```

```
130
131     return root;
132 }
133
134     private void s(View root){
135         TextView temp = (TextView)root.findViewById(R.id.texttemp_y);
136         ProgressBar tempBar =
137             ↪ (ProgressBar)root.findViewById(R.id.progressBary);
138         tempBar.setProgress(0);
139         temp.setText("-");

140         TextView hum = (TextView)root.findViewById(R.id.text_hum_y);
141         ProgressBar humBar =
142             ↪ (ProgressBar)root.findViewById(R.id.progressBarhy);
143         humBar.setProgress(0);
144         hum.setText("-");

145         TextView wei = (TextView)root.findViewById(R.id.text_wei_y);
146         ProgressBar weiBar =
147             ↪ (ProgressBar)root.findViewById(R.id.progressBarw_y);
148         weiBar.setProgress(0);
149         wei.setText("-");
150 }
```

5.3.1.7 Προβολή: Εβδομάδα

- FragmentWeek.java



Figure 96: Προβολή Δεδομένων - Εχθές

Πρόκειται για τη τρίτη προβολή μετρήσεων. Οι τιμές των μετρήσεων αφορούν τη τρέχουσα εβδομάδα. Ο μελισσοκόμος, μπορεί να έχει συγκεντρωτικά, όλες τις μετρήσεις της εβδομάδας.

```

1 package com.example.sofronas.ui.week;
2
3 import android.content.SharedPreferences;
4 import android.os.Bundle;
5 import android.view.Gravity;
6 import android.view.LayoutInflater;
```

```
7 import android.view.View;
8 import android.view.ViewGroup;
9 import android.widget.LinearLayout;
10 import android.widget.ProgressBar;
11 import android.widget.RelativeLayout;
12 import android.widget.TableLayout;
13 import android.widget.TableRow;
14 import android.widget.TextView;
15 import android.widget.Toast;

16
17 import androidx.annotation.NonNull;
18 import androidx.annotation.Nullable;
19 import androidx.constraintlayout.widget.ConstraintLayout;
20 import androidx.fragment.app.Fragment;
21 import androidx.lifecycle.Observer;
22 import androidx.lifecycle.ViewModelProvider;
23 import androidx.preference.PreferenceManager;

24
25 import com.android.volley.Request;
26 import com.android.volley.RequestQueue;
27 import com.android.volley.Response;
28 import com.android.volley.VolleyError;
29 import com.android.volley.toolbox.StringRequest;
30 import com.android.volley.toolbox.Volley;
31 import com.example.sofronas.R;

32
33 import org.w3c.dom.Text;
34
35 import java.util.ArrayList;
36 import java.util.Arrays;
37 import java.util.List;
```

```

39 public class FragmentWeek extends Fragment {
40
41     public View onCreateView(@NonNull LayoutInflater inflater,
42                             ViewGroup container, Bundle
43                             savedInstanceState) {
44         View root = inflater.inflate(R.layout.fragment_week,
45                                     container, false);
46         // s(root);
47
48         // Instantiate the RequestQueue.
49         RequestQueue queue = Volley.newRequestQueue(getActivity());
50         SharedPreferences settings =
51             PreferenceManager.getDefaultSharedPreferences(getActivity());
52         String ur = settings.getString("url", "127.0.0.1");
53         String us = settings.getString("username_id", "sof");
54         String url = "http://" + ur +
55         "/thesis/android/getWeek.php?us=" + us;
56
57         float textSize = 16;
58         // Request a string response from the provided URL.
59         StringRequest stringRequest = new
60             StringRequest(Request.Method.POST, url,
61                         new Response.Listener<String>() {
62                 @Override
63                 public void onResponse(String response) {
64                     // System.out.println("Response is: "+
65                     response);
66                     Boolean check = false;
67                     if(response.equals("")) {
68                         check = true;
69                         s(root);
70                     }
71                 }
72             });
73         queue.add(stringRequest);
74     }
75
76     @Override
77     public void onDestroy() {
78         super.onDestroy();
79         // s(root);
80     }
81 }

```

```

65             if(check == false) {
66                 String[] data =
67                   response.toString().split("<tr>");
68                     int start, end;
69                     List<String> daysOfWeek = new
70                         ArrayList<>();
71                         List<String> tempOfWeek = new
72                             ArrayList<>();
73                             List<String> humOfWeek = new
74                                 ArrayList<>();
75                                 List<String> weiOfWeek = new
76                                     ArrayList<>();
77                                     for (int i =1; i < data.length; i++){
78                                         start = data[i].indexOf("<td>") + 4;
79                                         end = data[i].indexOf("</td>");
80                                         String day =
81                                           data[i].substring(start,end);
82                                         daysOfWeek.add(day);
83                                         String[] rest =
84                                           data[i].split("<td>" + day + "</td>");
85                                         String[] res =
86                                           rest[1].split("</tr>");
87                                         String[] re = res[0].split("</td>");
88                                         String[] temp = re[0].split("<td>");
89                                         tempOfWeek.add(temp[1]);
90                                         String[] hum = re[1].split("<td>");
91                                         humOfWeek.add(hum[1]);
92                                         String[] wei = re[2].split("<td>");
93                                         weiOfWeek.add(wei[1]);
94                                     }
95                                     int x = daysOfWeek.size();

```

```
88                     ConstraintLayout main =
89             ↵     (ConstraintLayout)root.findViewById(R.id.constraint_week);
90                     TableLayout table =
91             ↵     (TableLayout)root.findViewById(R.id.table_week_id);
92                     //header row table
93                     TextView num = new
94             ↵     TextView(root.getContext());
95                     num.setText("#");
96
97                     num.setTextSize(textSize);
98
99                     TextView imerominia = new
100                ↵     TextView(root.getContext());
101                imerominia.setText("Ημερομηνία ");
102                imerominia.setTextSize(textSize);
103
104                     TextView therm = new
105                ↵     TextView(root.getContext());
106                therm.setText("Θερμοκρασία ");
107                therm.setTextSize(textSize);
108
109                     TextView ygras = new
110                ↵     TextView(root.getContext());
111                ygras.setText("Τρύπασια ");
112                ygras.setTextSize(textSize);
113
114                     TextView bar = new
115                ↵     TextView(root.getContext());
116                bar.setText("Βάρος");
117                bar.setTextSize(textSize);
```

```

112                     TableRow r = new
113             ↵   TableRow(getContext());
114                 r.addView(num);
115                 r.addView(imerominia);
116                 r.addView(therm);
117                 r.addView(ygras);
118                 r.addView(bar);
119                 r.setGravity(Gravity.CENTER);
120                 table.addView(r);

121 //           TableLayout table = new
122             ↵   TableLayout(getContext());
123                 for(int i=0; i<x; i++){
124                     TableRow row = new
125                         ↵   TableRow(getContext());
126                         TextView number = new
127                             ↵   TextView(getContext());
128                             number.setText(" " + (x-i) + " ");
129                             System.out.println(i);
130                             number.setTextSize(textSize);
131                             number.setGravity(Gravity.CENTER);

132                     TextView tv = new
133                         ↵   TextView(getContext());
134                         tv.setText(daysOfWeek.get(i) + " ");
135                         tv.setTextSize(textSize);
136                         tv.setGravity(Gravity.CENTER);

137                     char c = 0x00B0; //symbol of Celsius
138                     TextView tv1 = new
139                         ↵   TextView(getContext());

```

```

138                               tv1.setText("'" + tempOfWeek.get(i) +
139                               " " + c + " C");
140                               tv1.setTextSize(textSize);
141                               tv1.setGravity(Gravity.CENTER);
142
143                               TextView tv2 = new
144                               TextView(root.getContext());
145                               tv2.setText(" " + humOfWeek.get(i) +
146                               " % ");
147                               tv2.setTextSize(textSize);
148                               tv2.setGravity(Gravity.CENTER);
149
150                               TextView tv3 = new
151                               TextView(root.getContext());
152                               tv3.setText(" " + weiOfWeek.get(i) +
153                               " kg");
154                               tv3.setTextSize(textSize);
155                               tv3.setGravity(Gravity.CENTER);
156
157                               row.addView(number);
158                               row.addView(tv);
159                               row.addView(tv1);
160                               row.addView(tv2);
161                               row.addView(tv3);
162                               row.setGravity(Gravity.CENTER);
163                               table.addView(row);
}
//main.addView(table);
} else if(check == true){
    TextView nod =
        (TextView)root.findViewById(R.id.week_text_nodata);
    nod.setText(R.string.week_text_no_data);
}

```

```

164                     nod.setGravity(Gravity.CENTER);
165                 }
166             }
167         }, new Response.ErrorListener() {
168             @Override
169             public void onErrorResponse(VolleyError error) {
170                 System.out.println("Error: "+ error);
171                 System.out.println("That didn't work!");
172                 TextView nod =
173                     (TextView)root.findViewById(R.id.week_text_nodata);
174                     nod.setText(R.string.error);
175                     nod.setGravity(Gravity.CENTER);
176
177                     Toast t = Toast.makeText(root.getContext(),
178                     error.toString(), Toast.LENGTH_SHORT);
179                     t.show();
180                 }
181             });
182
183             // Add the request to the RequestQueue.
184             queue.add(stringRequest);
185             return root;
186         }
187
188         private void s(View root){
189             TextView nod =
190                 (TextView)root.findViewById(R.id.week_text_nodata);
191                 nod.setText(R.string.error);
192                 nod.setGravity(Gravity.CENTER);
193             }
194     }

```

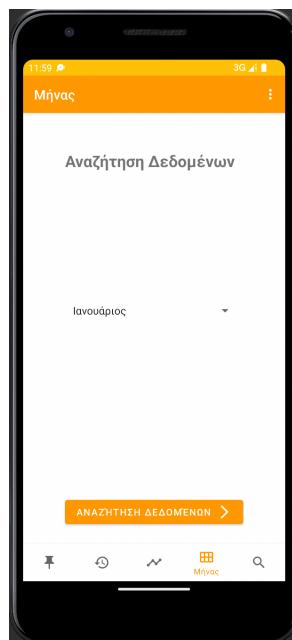
5.3.1.8 Προβολή: Μήνας

- **FragmentMonth.java** (Αρχείο επιλογής μήνα)
- **ShowDataMonth.java** (Τελικό αρχείο προβολής δεδομένων)
- **ListViewAdapter.java** (Μορφή προβολής)
- **Model.java** (Πρότυπο μορφοποιήσης πίνακα μετρήσεων)

Πρόκειται για τη τέταρτη προβολή μετρήσεων. Δίνεται η δυνατότητα στο χρήστη, να αναζητήσει μετρήσεις ενός συγκεκριμένου μήνα του τρέχοντος έτους με σκοπό τη προβολή όλων των μετρήσεων του μήνα. Με τη συγκεντρωτική προβολή, ο μελισσοκόμος έχει την δυνατότητα της πλήρης κατανόησης της εξέλιξης του μελισσοκομικού του λαού.

Η οργάνωση της προβολής χωρίζεται σε τέσσερα αρχεία. Το αρχείο **FragmentMonth.java** φορτώνεται πρώτο και περιέχει τη λίστα επιλογών των μηνών του έτους. Ο χρήστης επιλέγει το μήνα που επιθυμεί, στη περίπτωση που δεν διαλέξει συγκεκριμένο μήνα τότε αυτόματα ορίζεται ως επιλογή, ο μήνας Ιανουάριος. Στη συνέχεια, γίνεται η φόρτωση του αρχείου **ShowDataMonth.java** με την επιλογή του χρήστη.

Δημιουργείται ένα μοντέλο στο αρχείο **Model.java**, με σκοπό την διαμόρφωση των στηλών του πίνακα. Οι στήλες περιέχουν: ένα αύξων αριθμών, τη ημερομηνία, τη ψερμοκρασία, την υγρασία και το βάρος. Το αρχείο με τις μετρήσεις του μήνα συμπληρώνεται, εφόσον υπάρχουν αλλιώς εμφανίζεται ένα σχετικό μήνυμα για την απουσία δεδομένων. Για τη προβολή στην οθόνη τα δεδομένα χρησιμοποιούν, ένα πρότυπο λίστας το οποίο περιέχεται στο αρχείο **ListViewAdapter.java**.



(α') Προβολή μετρήσεων: Επιλογή Μήνα - Σχεδιασμός xml

N/N	Ημερομηνία	Θερμ/ία	Υγρασία	Βάρος
31	2022-01-31	3.29 °C	15.45 %	79.91 kg
30	2022-01-30	44.73 °C	13.92 %	119.04 kg
29	2022-01-29	21.96 °C	72.03 %	162.14 kg
28	2022-01-28	21.84 °C	50.92 %	147.84 kg
27	2022-01-27	21.82 °C	47.41 %	121.9 kg
26	2022-01-26	4.75 °C	27.18 %	162.57 kg
25	2022-01-25	42.54 °C	90.04 %	167.02 kg
24	2022-01-24	36.89 °C	80.56 %	80.67 kg
23	2022-01-23	29.36 °C	16.95 %	66.23 kg
22	2022-01-22	21.27 °C	22.73 %	45.3 kg
21	2022-01-21	33.05 °C	90.3 %	35.24 kg
20	2022-01-20	37.93 °C	90.35 %	131.56 kg
19	2022-01-19	38.55 °C	76.98 %	164.75 kg
18	2022-01-18	43.96 °C	76.26 %	38.82 kg
17	2022-01-17	36.18 °C	18.2 %	112.44 kg
16	2022-01-16	32.41 °C	49.83 %	164.64 kg
15	2022-01-15	18.61 °C	52.61 %	96.01 kg
14	2022-01-14	40.81 °C	75.36 %	158.8 kg
13	2022-01-13	34.93 °C	97.21 %	161.52 kg
12	2022-01-12	47.86 °C	49.68 %	152.35 kg
11	2022-01-11	34.73 °C	22.61 %	87.42 kg
10	2022-01-10	18.47 °C	69.38 %	77.19 kg
9	2022-01-09	24.03 °C	30.19 %	151.98 kg



(γ') Προβολή μετρήσεων: Μήνα - Σχεδιασμός xml

Σχήμα 97: Προβολή Δεδομένων - Μήνα

- Κώδικας - FragmentMonth.java

```
1 package com.example.sofronas.ui.month;
2
3 import androidx.lifecycle.ViewModelProvider;
4
5 import android.content.Context;
6 import android.content.Intent;
7 import android.os.Bundle;
8
9 import androidx.annotation.NonNull;
10 import androidx.annotation.Nullable;
11 import androidx.fragment.app.Fragment;
12
13 import android.view.LayoutInflater;
14 import android.view.View;
15 import android.view.ViewGroup;
16 import android.widget.AdapterView;
17 import android.widget.Button;
18 import android.widget.Spinner;
19 import android.widget.Toast;
20
21 import com.example.sofronas.MainActivity;
22 import com.example.sofronas.R;
23
24 public class FragmentMonth extends Fragment implements
25     View.OnClickListener {
26
27     public static FragmentMonth newInstance() {
28         return new FragmentMonth();
29     }
30
31     private Spinner spinner;
32
33     @Override
34     public void onCreate(Bundle savedInstanceState) {
35         super.onCreate(savedInstanceState);
36
37         spinner = findViewById(R.id.spinner);
38
39         spinner.setOnItemSelectedListener(this);
40
41         Intent intent = getIntent();
42
43         if (intent != null) {
44             String month = intent.getStringExtra("month");
45
46             if (!month.isEmpty()) {
47                 spinner.setSelection(month);
48             }
49         }
50
51         spinner.setAdapter(new ArrayAdapter(
52             requireContext(),
53             android.R.layout.simple_spinner_item,
54             android.R.id.text1,
55             new String[] {"January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"}));
56
57         spinner.setOnItemSelectedListener(this);
58
59     }
60
61     @Override
62     public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
63         View view = inflater.inflate(R.layout.fragment_month, container, false);
64
65         return view;
66     }
67
68     @Override
69     public void onClick(View v) {
70
71         int id = v.getId();
72
73         if (id == R.id.button) {
74
75             Intent intent = new Intent();
76
77             intent.putExtra("month", spinner.getSelectedItem().toString());
78
79             setResult(RESULT_OK, intent);
80
81             finish();
82         }
83     }
84 }
```

```
29
30     private Button searchMonthBtn;
31     private Spinner mySpinner;
32
33     @Override
34     public View onCreateView(@NonNull LayoutInflater inflater,
35     ↪ @Nullable ViewGroup container, @Nullable Bundle
36     ↪ savedInstanceState) {
37
38         View myView = inflater.inflate(R.layout.fragment_month,
39         ↪ container, false);
40
41         mySpinner = (Spinner)
42         ↪ myView.findViewById(R.id.list_of_months);
43         searchMonthBtn = (Button)
44         ↪ myView.findViewById(R.id.month_choose);
45         searchMonthBtn.setOnClickListener((View.OnClickListener)
46         ↪ this);
47
48         return myView;
49     }
50
51
52     //Call the showDataMonth
53     //If user doesn't provide any choice
54     //default month -> January
55     @Override
56     public void onClick(View v) {
57
58         String user_month = mySpinner.getSelectedItem().toString();
59         Intent myIntent = new Intent((MainActivity) getActivity(),
60         ↪ ShowDataMonth.class);
61
62         myIntent.putExtra("USER_MONTH_CHOICE", user_month);
63         int num = 0;
64
65
66         if(user_month.equals("Ιανουάριος")) {
67             num = 1;
```

```
54     }
55     if(user_month.equals("Φεβρουάριος")) {
56         num = 2;
57     }
58     if(user_month.equals("Μάρτιος")) {
59         num = 3;
60     }
61     if(user_month.equals("Απρίλιος")) {
62         num = 4;
63     }
64     if(user_month.equals("Μάιος")) {
65         num = 5;
66     }
67     if(user_month.equals("Ιούνιος")) {
68         num = 6;
69     }
70     if(user_month.equals("Ιούλιος")) {
71         num = 7;
72     }
73     if(user_month.equals("Αύγουστος")) {
74         num = 8;
75     }
76     if(user_month.equals("Σεπτέμβριος")) {
77         num = 9;
78     }
79     if(user_month.equals("Οκτώβριος")) {
80         num = 10;
81     }
82     if(user_month.equals("Νοέμβριος")) {
83         num = 11;
84     }
85     if(user_month.equals("Δεκέμβριος")) {
```

```

86             num = 12;
87         }
88         myIntent.putExtra("USER_MONTH_INT", num);
89         startActivity(myIntent);
90     }
91 }
```

- Κώδικας - ShowDataMonth.java

```

1 package com.example.sofronas.ui.month;
2
3 import androidx.appcompat.app.ActionBar;
4 import androidx.appcompat.app.AppCompatActivity;
5 import androidx.preference.PreferenceManager;
6
7 import android.app.Activity;
8 import android.content.SharedPreferences;
9 import android.os.Bundle;
10 import android.view.Gravity;
11 import android.widget.ArrayAdapter;
12 import android.widget.ListView;
13 import android.widget.TextView;
14 import android.widget.Toast;
15
16 import com.android.volley.Request;
17 import com.android.volley.RequestQueue;
18 import com.android.volley.Response;
19 import com.android.volley.VolleyError;
20 import com.android.volley.toolbox.StringRequest;
21 import com.android.volley.toolbox.Volley;
22 import com.example.sofronas.R;
```

```
23
24 import java.util.ArrayList;
25 import java.util.Arrays;
26 import java.util.List;
27
28 public class ShowDataMonth extends AppCompatActivity {
29     private ArrayList<Model> dataList;
30
31     @Override
32     protected void onCreate(Bundle savedInstanceState) {
33         super.onCreate(savedInstanceState);
34         setContentView(R.layout.activity_show_data_month);
35         String user_month_choice =
36             getIntent().getStringExtra("USER_MONTH_CHOICE");
37         int user_month_int =
38             getIntent().getIntExtra("USER_MONTH_INT",0);
39         ActionBar actionBar = getSupportActionBar();
40         if (actionBar != null) {
41             actionBar.setDisplayHomeAsUpEnabled(true);
42         }
43         //Set Logo at the top of bar
44         ActionBar actionBar2 = getSupportActionBar();
45         //set up title from a string
46         actionBar2.setTitle(user_month_choice);
47         actionBar2.setDisplayHomeAsUpEnabled(true);
48         actionBar2.setIcon(R.drawable.ic_month_activity);
49
50         System.out.println("From showdatamonth: " +
51             user_month_choice);
52         System.out.println("USER_MONTH_INT:" + user_month_int);
```

```

52         // Instantiate the RequestQueue.
53         RequestQueue queue = Volley.newRequestQueue(this);
54         SharedPreferences settings =
55             PreferenceManager.getDefaultSharedPreferences(this);
56         String ur = settings.getString("url", "127.0.0.1");
57         String us = settings.getString("username_id", "sof");
58         String url = "http://" + ur +
59             "/thesis/android/getMonth.php?m=" + user_month_int + "&us=" + us;
60         System.out.println(url);
61         // Request a string response from the provided URL.
62         StringRequest stringRequest = new
63             StringRequest(Request.Method.POST, url,
64                 new Response.Listener<String>() {
65                     @Override
66                     public void onResponse(String response) {
67                         System.out.println("Response is: "+
68                             response);
69                         Boolean check = false;
70                         if(response.equals("")) {
71                             check = true;
72                         }
73                         if(check == false) {
74                             String[] data =
75                                 response.toString().split("<tr>");
76                             System.out.println(data);
77                             int start, end;
78                             List<String> daysOfMonth = new
79                                 ArrayList<>();
80                             List<String> tempOfMonth = new
81                                 ArrayList<>();
82                             List<String> humOfMonth = new
83                                 ArrayList<>();
84                         }
85                     }
86                 });
87         queue.add(stringRequest);
88     }
89 }

```

```

76                         List<String> weiOfMonth = new
77                         ↵  ArrayList<>();
78
79                         for (int i =1; i < data.length; i++){
80                             start = data[i].indexOf("<td>") + 4;
81                             end = data[i].indexOf("</td>");
82                             String day =
83                               ↵  data[i].substring(start,end);
84                             daysOfMonth.add(day);
85
86                             String[] rest =
87                               ↵  data[i].split("<td>" + day + "</td>");
88                             String[] res =
89                               ↵  rest[1].split("</tr>");
90
91                             String[] re = res[0].split("</td>");
92                             String[] temp = re[0].split("<td>");
93                             tempOfMonth.add(temp[1]);
94                             String[] hum = re[1].split("<td>");
95                             humOfMonth.add(hum[1]);
96                             String[] wei = re[2].split("<td>");
97                             weiOfMonth.add(wei[1]);
98                         }
99
100
101                         dataList = new ArrayList<Model>();
102                         Model header = new
103                           ↵  Model("Ν/Ν", "Ημερομηνία", "Θερ/σία ", "Τγρασία", "Βάρος");
104                           dataList.add(header);
105
106
107                         char c = 0x00B0; //symbol of Celsius
108                         //make final data to import to list
109                         int x = daysOfMonth.size();
110                         for (int i=0; i<x; i++){

```

```

102                         Model temp = new
103             ↵ Model(Integer.toString(x-i), daysOfMonth.get(i),
104             ↵ tempOfMonth.get(i) + " " + c + "C", humOfMonth.get(i) + " %",
105             ↵ weiOfMonth.get(i) + " kg");
106
107             dataList.add(temp);
108         }
109
110     }
111
112     } else if(check == true){
113         Toast t =
114             ↵ Toast.makeText(getApplicationContext(),
115             ↵ R.string.no_data_yet_month, Toast.LENGTH_SHORT);
116             t.show();
117         }
118         System.out.println("check was:" + check);
119     }
120     }, new Response.ErrorListener() {
121         @Override
122         public void onErrorResponse(VolleyError error) {
123             Toast t = Toast.makeText(getApplicationContext(),
124             ↵ error.toString(), Toast.LENGTH_SHORT);
125             t.show();
126         }
127     });
128
129     // Add the request to the RequestQueue.
130     queue.add(stringRequest);

```

```

126     }
127
128     //Back Button
129     @Override
130     public boolean onSupportNavigateUp() {
131         finish();
132         return true;
133     }
134 }
```

- **Κώδικας - Model.java**

```

1 package com.example.sofronas.ui.month;
2
3 public class Model {
4
5     private String sNo;
6     private String date;
7     private String temp;
8     private String humd;
9     private String weight;
10
11    public Model(String sNo, String date, String temp, String humd,
12        ↪ String weight) {
13        this.sNo = sNo;
14        this.date = date;
15        this.temp = temp;
16        this.humd = humd;
17        this.weight = weight;
18    }
19
20    public String getsNo() {
21        return sNo;
22    }
23
24    public String getDate() {
```

```

24     return date;
25 }
26
27 public String getTemp() {
28     return temp;
29 }
30
31 public String getHumd() {
32     return humd;
33 }
34
35 public String getWeight(){
36     return weight;
37 }
38 }
```

- **Κώδικας - ListViewAdapter.java**

```

1 package com.example.sofronas.ui.month;
2
3 import android.app.Activity;
4 import android.view.LayoutInflater;
5 import android.view.View;
6 import android.view.ViewGroup;
7 import android.widget.BaseAdapter;
8 import android.widget.TextView;
9
10 import com.example.sofronas.R;
11
12 import java.util.ArrayList;
13
14 public class ListViewAdapter extends BaseAdapter {
15
16     public ArrayList<Model> dataList;
17     Activity activity;
18
19     public ListViewAdapter(Activity activity, ArrayList<Model>
→ dataList) {
```

```
20     super();
21     this.activity = activity;
22     this.dataList = dataList;
23 }
24
25 @Override
26 public int getCount() {
27     return dataList.size();
28 }
29
30 @Override
31 public Object getItem(int position) {
32     return dataList.get(position);
33 }
34
35 @Override
36 public long getItemId(int position) {
37     return position;
38 }
39
40 private class ViewHolder {
41     TextView mSNo;
42     TextView mDate;
43     TextView mTemp;
44     TextView mHumd;
45     TextView mWeight;
46 }
47
48 @Override
49 public View getView(int position, View convertView, ViewGroup
→ parent) {
50
51     ViewHolder holder;
52     LayoutInflator inflater = activity.getLayoutInflator();
53
54     if (convertView == null) {
55         convertView = inflater.inflate(R.layout.list_view_row,
→ null);
```

```
56         holder = new ViewHolder();
57         holder.mSNo = (TextView)
58             convertView.findViewById(R.id.sNo);
59         holder.mDate = (TextView)
60             convertView.findViewById(R.id.date);
61         holder.mTemp = (TextView)
62             convertView.findViewById(R.id.temp);
63         holder.mHumd = (TextView)
64             convertView.findViewById(R.id.humd);
65         holder.mWeight = (TextView)
66             convertView.findViewById(R.id.weight);
67         convertView.setTag(holder);
68     } else {
69         holder = (ViewHolder) convertView.getTag();
70     }
71
72     Model item = dataList.get(position);
73     holder.mSNo.setText(item.getsNo().toString());
74     holder.mDate.setText(item.getDate().toString());
75     holder.mTemp.setText(item.getTemp().toString());
76     holder.mHumd.setText(item.getHumd().toString());
77     holder.mWeight.setText(item.getWeight().toString());
78
79     return convertView;
80 }
81 }
```

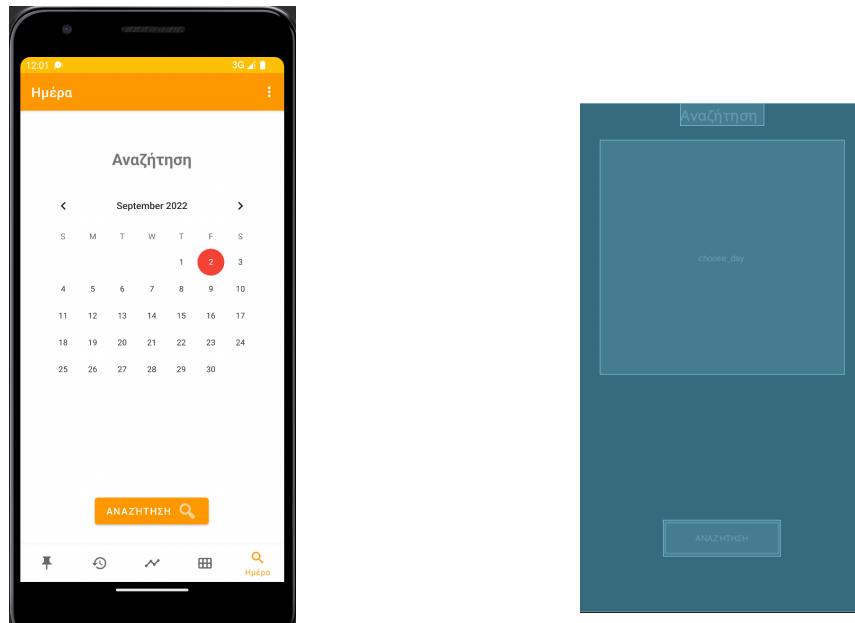
5.3.1.9 Προβολή: Αναζήτηση Ημέρας

- [FragmentSearch.java](#)
- [ShowDataDay.java](#)

Πρόκειται για τη πέμπτη και τελευταία προβολή μετρήσεων. Δίνεται η δυνατότητα στο χρήστη, να αναζητήσει μετρήσεις μιας συγκεκριμένης ημέρας. Ο χρήστης επιλέγει την ημέρα που επιθυμεί από το ημερολόγιο και επιλέγει αναζήτηση.

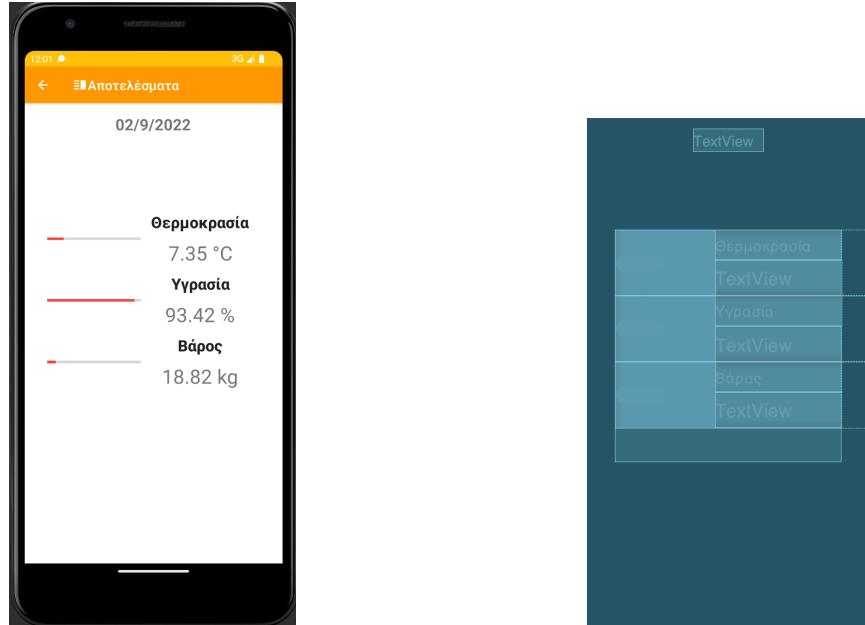
Στη συνέχεια, φορτώνεται μια προβολή με τις μετρήσεις της συγκεκριμένης ημέρας που επέλεξε. Στη περίπτωση που υπάρχουν μετρήσεις, θα εμφανιστούν κανονικά όπως θα φαινόταν στη προβολή σημερινής ημέρας.

Σε αντίθετη περίπτωση, έλλειψης μετρήσεων της συγκεκριμένης ημέρας, τότε ο χρήστης ενημερώνεται με αντίστοιχη προβολή όπου επικρατεί και η έλλειψη αριθμών.



(β') Προβολή μετρήσεων: Αναζήτηση Ημέρας

(α') Προβολή μετρήσεων: Αναζήτηση Ημέρας - Σχεδιασμός xml



(γ') Προβολή μετρήσεων: Συγκεκριμένης Η- (δ') Προβολή μετρήσεων: Συγκεκριμένης Η- μέρας

μέρας - Σχεδιασμός xml

Σχήμα 98: Προβολή Δεδομένων - Αναζήτηση Συγκεκριμένης Ημέρας

- Κώδικας - FragmentSearch.java

```
1 package com.example.sofronas.ui.search;
2
3 import androidx.lifecycle.ViewModelProvider;
4
5 import android.content.Context;
6 import android.content.Intent;
7 import android.os.Bundle;
8
9 import androidx.annotation.NonNull;
10 import androidx.annotation.Nullable;
11 import androidx.fragment.app.Fragment;
12
13 import android.view.LayoutInflater;
14 import android.view.View;
15 import android.view.ViewGroup;
16 import android.widget.Button;
17 import android.widget.CalendarView;
18 import android.widget.Toast;
19
20 import com.example.sofronas.MainActivity;
21 import com.example.sofronas.R;
22 import com.example.sofronas.ui.month.ShowDataMonth;
23
24 import java.sql.SQLOutput;
25 import java.text.ParseException;
26 import java.text.SimpleDateFormat;
27 import java.time.LocalDate;
28 import java.util.Calendar;
29 import java.util.Date;
30 import java.util.Locale;
31
32 public class FragmentSearch extends Fragment implements
33     ↪ View.OnClickListener {
34
35     private Button searchDayBtn;
36     private CalendarView calendar;
```

```

36     private String curDate;
37     private String curYear;
38     private String curMonth;
39     private String dateFromUser;
40
41     public static FragmentSearch newInstance() {
42         return new FragmentSearch();
43     }
44
45     @Override
46     public View onCreateView(@NonNull LayoutInflater inflater,
47         @Nullable ViewGroup container,
48             @Nullable Bundle savedInstanceState) {
49         View myView = inflater.inflate(R.layout.fragment_search,
50             container, false);
51         calendar = (CalendarView)
52             myView.findViewById(R.id.choose_day);
53         calendar.setOnDateChangeListener(new
54             CalendarView.OnDateChangeListener() {
55                 @Override
56                 public void onSelectedDayChange(CalendarView view, int
57                     year, int month, int dayOfMonth) {
58                     curDate = String.valueOf(dayOfMonth);
59                     curMonth = String.valueOf(month);
60                     curYear = String.valueOf(year);
61                     dateFromUser = curDate + "/" + curMonth + "/" +
62                         curYear;
63                 }
64             });
65
66         searchDayBtn = (Button) myView.findViewById(R.id.day_choose);
67         searchDayBtn.setOnClickListener((View.OnClickListener) this);
68         return myView;
69     }
70
71     //Call the showDataDay
72     //If user doesn't provide any choice
73     //default day -> today

```

```

68     @Override
69     public void onClick(View v) {
70         if(curDate == null && curMonth == null && curYear == null){
71             setToToday();
72         }
73         boolean moveOn = checkDate();
74         if (moveOn == true) {
75             Intent myIntent = new
76             ↪ Intent((MainActivity) getActivity(), ShowDataDay.class);
77             myIntent.putExtra("calendar_day_user", curDate);
78             myIntent.putExtra("calendar_month_user", curMonth);
79             myIntent.putExtra("calendar_year_user", curYear);
80             startActivity(myIntent);
81         } else {
82             Toast.makeText(getActivity(), "Cannot continue!"
83             ↪ ,Toast.LENGTH_SHORT).show();
84         }
85     }
86
87     //if user is not choose anything
88     //calendar view is set up default
89     //to today so from null setup day
90     public void setToToday(){
91         Calendar c = Calendar.getInstance();
92         SimpleDateFormat d = new SimpleDateFormat("dd/M/yyyy");
93         String strDate = d.format(c.getTime());
94
95         String[] textSplit = strDate.toString().split("/");
96         curDate = textSplit[0];
97         curMonth = Integer.toString(Integer.parseInt(textSplit[1]) -
98             ↪ 1);
99         curYear = textSplit[2];
100    }
101
102    public boolean checkDate() {
103        boolean good = true;
104        //compare dates > today

```

```

103     Calendar todayCal = Calendar.getInstance();
104     Calendar userCal = Calendar.getInstance();
105
106     Date dateToday = new Date();
107     todayCal.setTime(dateToday);
108
109     userCal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(curDate));
110     userCal.set(Calendar.MONTH, Integer.parseInt(curMonth));
111     userCal.set(Calendar.YEAR, Integer.parseInt(curYear));
112
113
114     Date dt = userCal.getTime();
115 //     System.out.println(dateToday);
116 //     System.out.println(dt);
117     userCal.setTime(dt);
118
119     if (userCal.after(todayCal)) {
120 //         System.out.println("User choice is after Today");
121 //         Toast.makeText(getApplicationContext(),"Are you come from future?"  

→ ,Toast.LENGTH_SHORT).show();
122         good = false;
123     }
124     return good;
125 }
126 }
```

- Κώδικας - ShowDataDay.java

```

1 package com.example.sofronas.ui.search;
2
3 import androidx.appcompat.app.ActionBar;
4 import androidx.appcompat.app.AppCompatActivity;
5 import androidx.preference.PreferenceManager;
6
7 import android.content.SharedPreferences;
8 import android.os.Bundle;
9 import android.widget.ProgressBar;
10 import android.widget.TextView;
```

```
11 import com.android.volley.Request;
12 import com.android.volley.RequestQueue;
13 import com.android.volley.Response;
14 import com.android.volley.VolleyError;
15 import com.android.volley.toolbox.StringRequest;
16 import com.android.volley.toolbox.Volley;
17 import com.example.sofronas.R;
18
19 public class ShowDataDay extends AppCompatActivity {
20
21     @Override
22     protected void onCreate(Bundle savedInstanceState) {
23         super.onCreate(savedInstanceState);
24         setContentView(R.layout.activity_show_data_day);
25
26         String calendar_day_user =
27             getIntent().getStringExtra("calendar_day_user");
28 //        System.out.println(calendar_day_user);
29
30         String calendar_month_user =
31             getIntent().getStringExtra("calendar_month_user");
32 //        System.out.println(calendar_month_user);
33
34         String calendar_year_user =
35             getIntent().getStringExtra("calendar_year_user");
36 //        System.out.println(calendar_year_user);
37
38         ActionBar actionBar = getSupportActionBar();
39         if (actionBar != null) {
40             actionBar.setDisplayHomeAsUpEnabled(true);
41         }
42         //Set Logo at the top of bar
43         ActionBar actionBar2 = getSupportActionBar();
44         //set up title from a string
45         actionBar2.setTitle(R.string.searched_day_results);
46         actionBar2setDisplayHomeAsUpEnabled(true);
47         actionBar2.setHomeAsUpIndicator(R.drawable.ic_month_activity);
```

```

46
47     String final_day = calendar_day_user + "/" +
48     ↵ (Integer.parseInt(calendar_month_user) + 1) + "/" +
49     ↵ calendar_year_user;
50
51     TextView date_top = findViewById(R.id.set_day_from_user_id);
52     date_top.setText(final_day);
53
54
55     // Instantiate the RequestQueue.
56     RequestQueue queue = Volley.newRequestQueue(this);
57     Sharedpreferences settings =
58     ↵ PreferenceManager.getDefaultSharedPreferences(this);
59     String ur = settings.getString("url", "127.0.0.1");
60     String us = settings.getString("username_id", "sof");
61     String url = "http://" + ur +
62     ↵ "/thesis/android/getSelectedDay.php";
63     //creating url specific for sql query
64     url = url + "?day=" + calendar_year_user + "/" +
65     ↵ (Integer.parseInt(calendar_month_user) + 1)+ "/" +
66     ↵ calendar_day_user;
67     url = url + "&us=" + us;
68
69     System.out.println("final url: " + url);
70     // Request a string response from the provided URL.
71     StringRequest stringRequest = new
72     ↵ StringRequest(Request.Method.GET, url,
73         new Response.Listener<String>() {
74             @Override
75             public void onResponse(String response) {
76                 // Display the first 500 characters of the
77                 ↵ response string.
78                 //System.out.println("Response is: "+
79                 ↵ response);
80
81             /* Scenario
82             empty string = no data today
83             data:
84                 first = temp

```

```

75                      second = hum
76                      third = weight
77      Analysis:
78          check if data are float
79          to fill bar with its int value
80      */
81      Boolean check = false;
82      if(response.equals("")) {
83          check = true;
84      }
85      if(check == false){
86          String[] data =
87              response.toString().split("<br>");
88          TextView temp =
89              (TextView)findViewById(R.id.texttemp);
90          ProgressBar tempBar =
91              (ProgressBar)findViewById(R.id.progressBart);
92          try {
93              Float te = Float.parseFloat(data[0]);
94              Integer k = te.intValue();
95              tempBar.setProgress(k);
96          } catch (NumberFormatException e) {
97
98              tempBar.setProgress(Integer.parseInt(data[0]));
99          }
100         temp.setText(data[0] + " " + (char)
101             0x00B0+"C");
102
103         TextView hum =
104             (TextView)findViewById(R.id.text_hum);
105         ProgressBar humBar =
106             (ProgressBar)findViewById(R.id.progressBarh);
107         try {
108             Float hu = Float.parseFloat(data[1]);
109             Integer h = hu.intValue();
110             humBar.setProgress(h);
111         } catch (NumberFormatException e){
112
113
114
115

```

```

106     ↪ humBar.setProgress(Integer.parseInt(data[1]));
107         }
108         hum.setText(data[1] + " %");
109
110         TextView wei =
111             ↪ (TextView)findViewById(R.id.text_wei);
112             ProgressBar weiBar =
113                 ↪ (ProgressBar)findViewById(R.id.progressBarw);
114                 try{
115                     Float tem =
116                         ↪ Float.parseFloat(data[2]);
117                         Integer i = tem.intValue();
118                         weiBar.setProgress(i);
119                     } catch (NumberFormatException e) {
120
121             ↪ weiBar.setProgress(Integer.parseInt(data[2]));
122                 }
123                 wei.setText(data[2] + " kg");
124             } else {
125                 TextView temp =
126                     ↪ (TextView)findViewById(R.id.texttemp);
127                     ProgressBar tempBar =
128                         ↪ (ProgressBar)findViewById(R.id.progressBart);
129                         tempBar.setProgress(0);
130                         temp.setText("–");
131
132                 TextView hum =
133                     ↪ (TextView)findViewById(R.id.text_hum);
134                     ProgressBar humBar =
135                         ↪ (ProgressBar)findViewById(R.id.progressBarh);
136                         humBar.setProgress(0);
137                         hum.setText("–");
138
139                 TextView wei =
140                     ↪ (TextView)findViewById(R.id.text_wei);
141                     ProgressBar weiBar =
142                         ↪ (ProgressBar)findViewById(R.id.progressBarw);
143                         weiBar.setProgress(0);

```

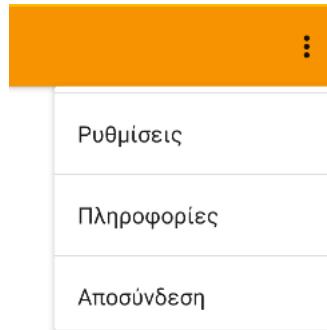
```
133                         wei.setText("-");  
134                     }  
135                 }  
136             }, new Response.ErrorListener() {  
137                 @Override  
138                 public void onErrorResponse(VolleyError error) {  
139                     System.out.println("Error: " + error);  
140                     System.out.println("That didn't work!");  
141                 }  
142             });  
143  
144             // Add the request to the RequestQueue.  
145             queue.add(stringRequest);  
146         }  
147  
148     @Override  
149     public boolean onSupportNavigateUp() {  
150         finish();  
151         return true;  
152     }  
153 }
```

5.3.1.10 Μενού Επιλογών

- [SettingsActivity.java](#)
- [InfoActivity.java](#)
- [LoginActivity.java](#)

Στη πάνω δεξιά μεριά, υπάρχει μενού το οποίο αναδύεται με το πάτημα των τριών τελειών. Δίνει στο χρήστη επιλογές για τη προβολή των ρυθμίσεων, τη προβολή πληροφοριών για την εφαρμογή αλλά και επιλογή αποσύνδεσης του χρήστη από την εφαρμογή.

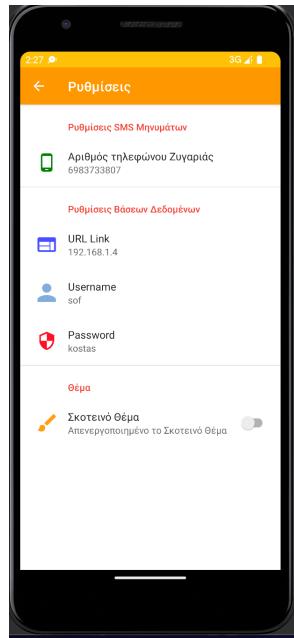
Στη περίπτωση που ο χρήστης επιλέξει να κάνει αποσύνδεση από το σύστημα, τότε καλείται η προβολή είσοδου στο σύστημα τερματίζοντας τη σύνδεση του.



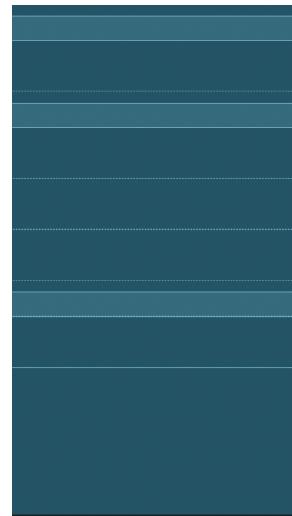
Σχήμα 99: Μενού επιλογών - Επάνω δεξιά

5.3.1.11 Προβολή: Ρυθμίσεις

- **SettingsActivity.java**



(α') Προβολή Ρυθμίσεις



(β') Προβολή Ρυθμίσεις - Σχεδιασμός xml

Σχήμα 100: Προβολή Ρυθμίσεις

Στις ρυθμίσεις, ο χρήστης μπορεί να ορίσει τον αριθμό της ηλεκτρονικής ζυγαριάς, με σκοπό η τηλεματική εφαρμογή να διαβάζει μόνο από αυτόν τον αριθμό τα γραπτά μηνύματα. Ακόμη, μπορεί να ορίσει, τις ρυθμίσεις του server πάντα μετά από υπόδειξη του προγραμματιστή και τέλος μπορεί να επιλέξει ανάμεσα σε φωτεινό ή σκούρο θέμα.

5.3.1.12 Προβολή: Πληροφορίες

- InfoActivity.java



(α') Προβολή Πληροφορίες



(β') Προβολή Πληροφορίες - Σχεδιασμός xml

Σχήμα 101: Προβολή Πληροφοριών

Στη συγκεκριμένη προβολή παρουσιάζονται στον χρήστη κάποια στοιχεία επικοινωνίας με τον διαχειριστή της εφαρμογής.

```

1 package com.example.sofronas;
2
3 import android.os.Bundle;
4
5 import androidx.appcompat.app.ActionBar;
6 import androidx.appcompat.app.AppCompatActivity;
7

```

```
8 public class InfoActivity extends AppCompatActivity {  
9  
10    @Override  
11    protected void onCreate(Bundle savedInstanceState) {  
12        super.onCreate(savedInstanceState);  
13        setContentView(R.layout.info);  
14  
15        ActionBar actionBar = getActionBar();  
16        if (actionBar != null) {  
17            actionBar.setDisplayHomeAsUpEnabled(true);  
18        }  
19        //Set Logo at the top of bar  
20        ActionBar actionBar2 = getActionBar();  
21        //set up title from a string  
22        actionBar2.setTitle(getString(R.string.info_app));  
23        actionBar2.setDisplayHomeAsUpEnabled(true);  
24    }  
25  
26    //Back Button  
27    @Override  
28    public boolean onSupportNavigateUp() {  
29        finish();  
30        return true;  
31    }  
32 }
```

Η προβολή, περιέχει τρεις εικόνες και τρία κείμενα για τη παρουσίαση πληροφοριών.

5.3.1.13 Αρχείο Ενημέρωσης Δεδομένων Βάσης Δεδομένων

- **SmsReceiver.java**

Το αρχείο, είναι υπεύθυνο για το διάβασμα των μηνυμάτων από τον αριθμό που έχει αντιστοιχηθεί στην ηλεκτρονική ζυγαριά. Η μορφή του γραπτού μηνύματος που λαμβάνει το κινητό του μελισσοκόμου είναι η εξής:

Μορφή SMS:

Thermokrasia: 22.13 C
Ygrasia: 43 %
Weight: 34.03 kg

Όταν έρθει το μήνυμα, εκτελείται ο αλγόριθμος που διαβάζει κατάλληλα το μήνυμα για το ξεχωρισμό των σημαντικών πληροφοριών, δηλαδή, τις τιμές από τους αισθητήρες.

Στη συνέχεια, δημιουργείται ένας σύνδεσμος, για την ενημέρωση της βάσης δεδομένων με τις νέες τιμές που έλαβε η τηλεματική εφαρμογή.

Ο σύνδεσμος:

<http://domain.gr/thesis/android/handler.php?user=sof&thermokrasia=30...&ugrasisa=10&baros=29.7>

Για τη δοκιμή του, απαιτείται η τοπική διεύθυνση IP που κατέχει ο υπολογιστής στη θέση του domain.gr.

```

1 package com.example.sofronas.sms;
2
3 import android.content.BroadcastReceiver;
4 import android.content.Context;
5 import android.content.Intent;
6 import android.content.SharedPreferences;
7 import android.os.Bundle;
8 import android.telephony.SmsMessage;
9 import android.widget.ListView;
10 import android.widget.Toast;
11
12 import androidx.preference.PreferenceManager;
13
14 import com.android.volley.Request;
15 import com.android.volley.RequestQueue;
```

```

16 import com.android.volley.Response;
17 import com.android.volley.VolleyError;
18 import com.android.volley.toolbox.StringRequest;
19 import com.android.volley.toolbox.Volley;
20 import com.example.sofronas.R;
21 import com.example.sofronas.SettingsActivity;
22 import com.example.sofronas.ui.month.ListViewAdapter;
23 import com.example.sofronas.ui.month.Model;
24 import com.example.sofronas.ui.month.ShowDataMonth;
25
26 import org.json.JSONObject;
27
28 import java.util.ArrayList;
29 import java.util.List;
30
31 public class SmsReceiver extends BroadcastReceiver {
32
33     public void onReceive(Context context, Intent intent) {
34         String s3 = "android.provider.Telephony.SMS_RECEIVED";
35         if(intent.getAction().equals(s3)){
36             Bundle bundle = intent.getExtras(); //---get
37             ↪ the SMS message passed in---
38             SmsMessage[] msgs = null;
39             String msg_from;
40             String phoneNumber = SettingsActivity.getPhone(context);
41             String username = SettingsActivity.getUsername(context);
42             String password = SettingsActivity.getPassword(context);
43             String host = SettingsActivity.getHost(context);
44             if (bundle != null){
45                 //---retrieve the SMS message received---
46                 try{
47                     Object[] pdus = (Object[]) bundle.get("pdus");
48                     msgs = new SmsMessage[pdus.length];
49                     for(int i=0; i<msgs.length; i++){
50                         msgs[i] =
51                         ↪ SmsMessage.createFromPdu((byte[])pdus[i]);
52                         msg_from = msgs[i].getOriginatingAddress();
53                         String msgBody = msgs[i].getMessageBody();

```

```

52
53         if(msg_from.equals(phoneNumber)){
54             System.out.println(msgBody);
55             Toast.makeText(context, msg_from + ":" +
56             ↪ +msgBody, Toast.LENGTH_SHORT).show();
57             String[] bodies = msgBody.split("C");
58             String thermString = bodies[0];
59             String[] temp = thermString.split(":");
60             thermString = temp[1];
61
62             String ugrasString = bodies[1];
63             temp = ugrasString.split("%");
64             String temp1 = temp[1]; //Weight
65             ugrasString = temp[0];
66             temp = ugrasString.split(":");
67             ugrasString = temp[1];
68
69             String[] kg = temp1.split(" kg");
70             String temp2[] = kg[0].split(":");
71             String kgString = temp2[1];
72
73             double thermokrasia =
74             ↪ Double.parseDouble(thermString);
75             double ugrasia =
76             ↪ Double.parseDouble(ugrasString);
77             double baros =
78             ↪ Double.parseDouble(kgString);
79
80             // Instantiate the RequestQueue.
81             RequestQueue queue =
82             ↪ Volley.newRequestQueue(context);
83             SharedPreferences settings =
84             ↪ PreferenceManager.getDefaultSharedPreferences(context);
85             String user =
86             ↪ settings.getString("username", "null");
87             String ur = settings.getString("url",
88             ↪ "127.0.0.1");
89             String url = "http://" + ur +

```

```

82     ↪ "/thesis/android/handler.php?user=" + user;
83             String urlString = url +
84             "&thermokrasia=" + thermokrasia + "&ugrasia=" + ugrasia +
85             "&baros=" + baros;
86
87             // Request a string response from the
88             ↪ provided URL.
89             StringRequest stringRequest = new
90             ↪ StringRequest(Request.Method.POST, urlString,
91                         new Response.Listener<String>() {
92                             @Override
93                             public void
94                             ↪ onResponse(String response) {
95
96                             ↪ System.out.println("Response is: " + response);
97                             Boolean check = false;
98                             if(response.equals("")) {
99                                 check = true;
100                            }
101                            }
102                            , new Response.ErrorListener() {
103                                @Override
104                                public void
105                                ↪ onErrorResponse(VolleyError error) {
106                                    Toast t =
107                                    ↪ Toast.makeText(context, error.toString(), Toast.LENGTH_SHORT);
108                                    t.show();
109                                }
110                            });
111
112             // Add the request to the RequestQueue.
113             queue.add(stringRequest);
114         }
115     }
116     }catch(Exception e){
117         e.printStackTrace();
118         System.out.println("ERROR: SMS: " + e);
119     }
120 }
```

```

111     }
112     else {
113         Toast.makeText(context, "No access to SMS",
114             ↪ Toast.LENGTH_SHORT).show();
114     }
115 }
116 }
```

Σε περίπτωση που ο χρήστης δεν έχει δώσει τη κατάλληλη άδεια στη τηλεματική εφαρμογή να διαβάζει τα γραπτά μηνύματα SMS τότε υπάρχει αδυναμία διαβάσματος και ακολουθεί αποτυχία στην ενημέρωση του συστήματος.

Ο χρήστης ενημερώνεται καταλλήλως. Άδεια στην εφαρμογή μπορεί να οριστεί και μετά την εγκατάσταση της, προκειμένου να λυθεί το πρόβλημα αδειοδότησης της εφαρμογής.

Ο χρήστης, πηγαίνει στις 'Ρυθμίσεις' → 'Εφαρμογές' → 'Τηλεματική Εφαρμογή' → 'Άδειες' → 'Πρόσβαση στα γραπτά μηνύματα' → 'Αποδοχή'

5.3.2 Δομή Αρχείων Επικοινωνίας Εφαρμογής με την Βάση Δεδομένων

Για την επικοινωνία της τηλεματικής εφαρμογής με τη βάση δεδομένων υπάρχει μέρος στην ιστοσελίδα που συμβάλει σε αυτή.

```
/  
└── android .... Φάκελος με αρχεία που απαιτούνται από τη τηλεματική εφαρμογή  
    ├── getMonth.php  
    ├── getSelectedDay.php  
    ├── getToday.php  
    ├── getWeek.php  
    └── handler.php
```

- **getMonth.php**

Το αρχείο, κάνει αναζήτηση στη βάση δεδομένων για τις μετρήσεις του μήνα που επέλεξε ο χρήστης. Η εφαρμογή, δίνει στο αρχείο το μήνα που επέλεξε ο χρήστης και ακόμη το όνομα χρήστη, με σκοπό να προβληθούν τα δεδομένα του συγκεκριμένου χρήστη.

```
1 <?php  
2     require ' ../ settings . php ' ;  
3  
4     $conn = new mysqli ( DB_HOST , DB_USER , DB_PASS , DB_NAME ) ;  
5     $current_year = date ( " Y " ) ;  
6  
7     $sql = " SELECT * FROM data  
8             WHERE MONTH( imerominia ) = " . $_GET [ ' m ' ] .  
9                 " AND YEAR( imerominia ) = " . $current_year . " AND user  
= " . $_GET [ " us " ] . " "  
10            ORDER BY imerominia DESC " ;  
11  
12     $monday = strtotime ( ' monday this week ' ) ;  
13     $sunday = strtotime ( ' sunday this week ' ) ;  
14     $start = date ( " d/m/Y " , $monday ) ;  
15     $last = date ( " d/m/Y " , $sunday ) ;  
16  
17     $result = mysqli_query ( $conn , $sql ) ;  
18     if ( mysqli_num_rows ( $result ) > 0 ) {  
         echo '<table>
```

```
20      <tbody>
21          ';
22      while ($row = mysqli_fetch_array($result)){
23          echo "<tr>";
24              echo "<td>". $row[ 'imerominia ']. "</td>" ;
25              echo "<td>". $row[ 'thermokrasia ']. "</td>" ;
26              echo "<td>". $row[ 'ugrasia ']. "</td>" ;
27              echo "<td>". $row[ 'baros ']. "</td>" ;
28          echo "</tr>";
29      }
30      echo "</tbody>
31          </table>";
32  }
33  mysqli_close($conn);
34
35?>
```

- [getSelectedDay.php](#)

Το αρχείο, κάνει αναζήτηση στη βάση δεδομένων για τις μετρήσεις μιας συγκεκριμένης ημέρας, που επέλεξε ο χρήστης. Η εφαρμογή, δίνει στο αρχείο την ημέρα σε μορφή ημερομηνίας, που επέλεξε ο χρήστης και ακόμη το όνομα χρήστη, με σκοπό να προβληθούν τα δεδομένα του συγκεκριμένου χρήστη.

```

1 <?php
2     require '../settings.php';
3
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     $d = $_GET['day'];
6     $sql = "SELECT * FROM data
7             WHERE imerominia = '". $d . "'"
8             AND user = '". $_GET["us"] . "'";
9     $result = mysqli_query($conn, $sql);
10    if (mysqli_num_rows($result) > 0) {
11        $row = mysqli_fetch_row($result);
12        echo $row[3];
13        echo "<br>";
14        echo $row[4];
15        echo "<br>";
16        echo $row[5];
17    } else {
18        $row = NULL;
19    }
20
21    mysqli_close($conn);
22
23 ?>
```

- **getToday.php**

Πρόκεται για το αρχείο το οποίο κάνει αναζήτηση στη βάση δεδομένων για τις μετρήσεις της σημερινής ημέρας. Η εφαρμογή δίνει στο αρχείο, το όνομα χρήστη με σκοπό να γίνουν προβολή τα δεδομένα του συγκεκριμένου χρήστη.

```

1 <?php
2     require '../settings.php';
3
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     $today = date("Y-m-d");
6     $sql = "SELECT * FROM data
7             WHERE imerominia = '". $today . "'
8                 AND user = '" . $_GET[ "us" ] . "' ";
9     $result = mysqli_query($conn, $sql);
10    if (mysqli_num_rows($result) > 0) {
11        $row = mysqli_fetch_row($result);
12        echo $row [3];
13        echo "<br>";
14        echo $row [4];
15        echo "<br>";
16        echo $row [5];
17    } else {
18        $row = NULL;
19    }
20
21    mysqli_close($conn);
22
23 ?>
```

- **getWeek.php**

Το αρχείο, κάνει αναζήτηση στη βάση δεδομένων για τις μετρήσεις της τρέχουσας εβδομάδας. Η εφαρμογή δίνει στο αρχείο, το όνομα χρήστη, με σκοπό να προβληθούν τα δεδομένα του συγκεκριμένου χρήστη.

```

1 <?php
2     require '../settings.php';
3
4     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
5     $today = date("Y-m-d");
6     $sql = "SELECT * FROM data
7             WHERE YEARWEEK(imerominia) = YEARWEEK(NOW())
8                 AND user = '".$_GET["us"]."'
9                 ORDER BY imerominia DESC";
10
11    $monday = strtotime('monday this week');
12    $sunday = strtotime('sunday this week');
13    $start = date("d/m/Y", $monday);
14    $last = date("d/m/Y", $sunday);
15
16    $result = mysqli_query($conn, $sql);
17    if (mysqli_num_rows($result) > 0) {
18        echo '<table>
19              <tbody>
20                  ';
21        while ($row = mysqli_fetch_array($result)){
22            echo "<tr>";
23            echo "<td>".$row['imerominia']. "</td>";
24            echo "<td>".$row['thermokrasia']. "</td>";
25            echo "<td>".$row['ugrasia']. "</td>";
26            echo "<td>".$row['baros']. "</td>";
27            echo "</tr>";
28        }
29        echo "</tbody>
30              </table>';
31    }
32    mysqli_close($conn);
33
34 ?>
```

- **handler.php**

Το αρχείο, χρησιμοποιείται για να ενημερώνει τη βάση δεδομένων με τις νέες τιμές, που η τηλεματική εφαρμογή έχει διαβάσει από το γραπτό μήνυμα που στέλνει η ηλεκτρονική ζυγαριά.

```

1 <?php
2     // connect to database
3     require "../settings.php";
4
5     $conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
6
7     if (!$conn) {
8         die("Connection failed: " . mysqli_connect_error());
9         echo "failed";
10    }
11
12    $imer = date("Y-m-d");
13    // $imer = date('Y-m-d', strtotime($imer));
14    echo $imer . "<br>";
15    $user = $_GET['user'];
16    $ther = $_GET['thermokrasia'];
17    $ugr = $_GET['ugrasia'];
18    $bar = $_GET['baros'];
19
20
21    print_r($_REQUEST);
22    $sql = "INSERT INTO data (user, imerominia, thermokrasia,
23    ugrasia, baros)
24        VALUES ('" . $user . "', '" . $imer . "', '" . $ther .
25        "', '" . $ugr . "', '" . $bar . "')";
26    echo $sql;
27    $result = mysqli_query($conn, $sql);
28
29    mysqli_close($conn);
?>
```

Κεφάλαιο 6

Συμπεράσματα / Μελλοντικές Επεκτάσεις

6.1 Συμπεράσματα

Κατά τη διάρκεια εκπόνησης της πτυχιακής εργασίας, παρατηρήθηκε ότι τα έξοδα ενός μελισσοκόμου μειώθηκαν κατά 30%. Αυτό οφείλεται, στη συχνή παρακολούθηση (απομακρυσμένα) της πορείας των μελισσών.

Μέσω των δεδομένων που συλλέγονται καθημερινά, ο μελισσοκόμος μπορεί να ορίσει τη συχνότητα της παρουσίας του στο χώρο. Αυτό έχει ως αποτέλεσμα, τη μείωση των άσκοπων μετακινήσεων του, αλλά και την άμεση επέμβαση του σε περιπτώσεις ανάγκης.

Με αυτό το τρόπο μπορούν να προληφθούν ζημιές στα μελίσσια, οι οποίες θα έχουν σοβαρό οικονομικό αντίκτυπο στο προϋπολογισμό του μελισσοκόμου.

6.2 Μελλοντικές Επεκτάσεις

Μετά την αρχική υλοποιήση του συστήματος, παρατηρήθηκε ότι θα μπορούσε να γίνουν κάποιες προσθήκες με σκοπό την εξέλιξη και τη βελτιστοποίηση του. Μερικές από αυτές είναι:

- Arduino

- Προσθήκη αισθητήρων θερμοκρασίας, υγρασίας, βάρους και σε άλλες κυψέλες. Συγκέντρωση δεδομένων σε μια κεντρική κυψέλη και αποστολή των δεδομένων για περισσότερες από μια κυψέλη.
- Προσθήκη αισθητήρα εντοπισμού χραδασμού, με συνεργασία ενός αισθητήρα GPS, για την άμεση ενημέρωση του μελισσοκόμου στη περίπτωση κλοπής του λαού, καθώς παρατηρείται συχνό φαινόμενο κλοπής κυψελών.
- Προσθήκη κάμερας, για τη λήψη φωτογραφιών σε περίπτωση εντοπισμού κίνησης μεγαλόσωμων ζώων αλλά και ανθρώπων.

- Website

- Προσθήκη προβολής δεδομένων, από δεύτερη ή περισσότερες ηλεκτρονικές ζυγαριές για παρουσίαση.
- Προβολή φωτογραφιών μετά από εντοπισμό κίνησης.
- Προσθήκη στίγματος GPS στις προβολές δεδομένων.

- Android
 - Καταγράφη δεδομένων από δύο ή περισσότερες ηλεκτρονικές ζυγαριές στη βάση δεδομένων.
 - Προσθήκη προβολής δεδομένων από περισσότερες ηλεκτρονικές ζυγαριές.
 - Προσθήκη δυνατότητας υπενθύμισης για ανανέωση πακέτου γραπτών μηνυμάτων και πληρωμή λογαριασμού.

Αναφορές

Internet:

- [1] <https://el.wikipedia.org/wiki/Μελισσοκομία>
- [2] <https://docs.arduino.cc/hardware/uno-rev3>
- [3] <http://users.sch.gr/manpoul/docs/arduino/ProgrammingArduino.pdf>
- [4] <https://docs.arduino.cc/retired/getting-started-guides/ArduinoGSMShield>
- [5] <https://store.arduino.cc/products/arduino-uno-rev3>
- [6] <https://www.arduino.cc/reference/en/libraries/>
- [7] <https://www.arduino.cc/reference/en/libraries/category/sensors/>
- [8] <https://www.arduino.cc/reference/en/libraries/dht-sensor-library/>
- [9] <https://www.arduino.cc/reference/en/libraries/ds18b20/>
- [10] <https://www.arduino.cc/reference/en/libraries/hx711-arduino-library/>
- [11] <https://developer.android.com/docs>
- [12] <https://stackoverflow.com/questions/2198410/how-to-change-title-of-activity-in-android>
- [13] <https://stackoverflow.com/questions/7326155/align-text-center-with-android>
- [14] <https://stackoverflow.com/questions/35699540/android-studio-how-to-close-quit-exit-kill-etc-an-application>
- [15] <https://techicaltutorial.blogspot.com/2020/03/android-studio-listview-multi-column-example.html>
- [16] <https://developer.android.com/guide/topics/ui/notifiers/toasts>
- [17] <https://google.github.io/volley/simple>
- [18] <https://stackoverflow.com/questions/7089313/android-listen-for-incoming-sms-messages>

- [19] <https://stackoverflow.com/questions/39112632/changing-background-color-in-a-activity>
- [20] <https://stackoverflow.com/questions/46031503/android-image-button-to-close-popup>
- [21] <https://stackoverflow.com/questions/6014028/closing-application-with-exit-button/6014093>
- [22] <https://google-developer-training.github.io/android-developer-fundamentals-course-practicals/en/Unit>
- [23] <https://stackoverflow.com/questions/39476429/preference-settings-in-android-studio-how-to-connect-the-value-in-edittextprefer>
- [24] <https://stackoverflow.com/questions/47495534/how-to-enable-night-mode-programmatically>
- [25] <https://www.section.io/engineering-education/bottom-navigation-bar-in-android/>
- [26] <https://medium.com/@son.rommer/fix-cleartext-traffic-error-in-android-9-pie-2f4e9e2235e6>
- [27] <https://stackoverflow.com/questions/6343166/how-can-i-fix-android-os-networkonmainthreadexception>
- [28] <https://stackoverflow.com/questions/8654876/http-get-using-android-httpurlconnection>
- [29] <https://stackoverflow.com/questions/33229869/get-json-data-from-url-using-android/35916046>
- [30] <https://stackoverflow.com/questions/42176506/how-to-get-a-php-variable-back-to-android-app>
- [31] <https://github.com/RamiJ3mli/PercentageChartView>
- [32] <https://github.com/coliff/dark-mode-switch>
- [33] <https://stackoverflow.com/questions/12637693/android-calling-activity-from-fragment>

- [34] <https://stackoverflow.com/questions/2091465/how-do-i-pass-data-between-activities-in-android-application>
- [35] <https://stackoverflow.com/questions/1947933/how-to-get-spinner-value>
- [36] <https://code.tutsplus.com/tutorials/how-to-add-a-dropdown-menu-in-android-studio--cms-37860>
- [37] <https://stackoverflow.com/questions/42001007/android-studio-back-button-for-previous-activity>
- [38] <https://stackoverflow.com/questions/11419141/fading-in-text-in-android-using-animationutils-loadanimation/14152152>
- [39] <https://stackoverflow.com/questions/37411696/volley-is-not-working-when-loading-from-xampp-local-server>
- [40] <https://www.javatpoint.com/android-volley-library-registration-login-logout>
- [41] <https://www.apachefriends.org/docs/>
- [42] <https://www.php.net/manual/en/>
- [43] <https://dev.w3.org/html5/spec-LC/>
- [44] <https://devdocs.io/css/>
- [45] <https://getbootstrap.com/docs/5.2/getting-started/introduction/>
- [46] <https://devdocs.io/javascript/>
- [47] <https://stackoverflow.com/questions/17150171/page-redirect-after-x-seconds-wait-using-javascript>
- [48] <https://dcodemania.com/post/responsive-login-registration-form-design-bootstrap-jquery>
- [49] <https://github.com/coliff/bootstrap-show-password-toggle>
- [50] https://www.w3schools.com/sql/func_mysql_yearweek.asp

Ebooks:

- [51] Simon Monk, Arduino + Android Projects for the Evil Genius: Control Arduino with your Smartphone or Tablet (2011)
- [52] Brad Kendall, makeuseof getting started with arduino - a beginner's guide (2013)
- [53] Lady Ada, Arduino Tips Tricks and Techinques (2021)
- [54] Tony Olsson, David Gaetanom, Jonas Odhner, Samson Wiklund, Open Soft- wear fashionable prototyping and wearable computing using the Arduino (2008)
- [55] Guy W. Lecky-Thompson, Just enough web programming with XHTML,PHP and MYSQL (2008)
- [56] W.Jason Gilmore, Robert H. Treat, Beginning PHP and PostgreSQL 8, From Novice to Professional (2006)
- [57] Robert Richards, Pro PHP XML and Web Services (2006)
- [58] Robin Nixon, Learning PHP, MySQL, JavaScript and CSS , Second Edition (2012)
- [59] Rob Crowther, Hello! HTML & CSS3 A user-friendly reference guide (2013)
- [60] Simon Collison, Beginning CSS Web Development From Novice To Profes- sional (2006)
- [61] J.F. Di Marzio, Beginning Android Programmaing with Android Studio, 4th edition (2017)
- [62] Adam Gerber - Clifton Craig, Learn Android Studio - Build Android Apps Quickly and Effectively (2015)
- [63] Barbar Hohensee, Getting Started with Android Studio (2014)
- [64] J.Paul Cardle, Android App Development in Android Studio, Java + Android Edition for Beginners
- [65] Joseph Annuzzi Jr., Lauren Darcey and Shane Conder,Introduction to An- droid Application Development: Android Essentials,Addison-Wesley Profes- sional (2013)

- [66] Neil Smyth, Android Studio Development Essentials, CreateSpace Independent Publishing Platform (2016)
- [67] Sam Key, Android Programming in a Day, CreateSpace Independent Publishing Platform (2015)
- [68] Barry A. Burd, Android Application Development All-in-One For Dummies, For Dummies (2015)
- [69] Herbert Schildt, Java: A Beginner's Guide, 7th Edition (2017)

Βιβλία:

- [70] Βασιλική Λιάκου, Διαχείριση Μελισσοκομείου για εντατική παραγωγή μελιού (1995), Θεσσαλονίκη, σ:23-31
- [71] Πογαρίδης Δημήτρης, Ενσωματώμενα Συστήματα, οι Μικροελεγκτές AVR και Arduino , σ:3,18,79,137,201,333
- [72] Φίλιππος Ψύχαλος και Σία Εκδοτική Ο.Ε. Εκδόσεις, Σύγχρονη Μελισσοκομία (2007), σ: 25
- [73] Νικ. Ι. Νικολαΐδη, Μελισσοκομία, Σύγχρονες Μέθοδοι Ενταντικής Εκμεταλευσης, Έκδοση 8η (2005), σ:70,73,78,79,83,87-91